

MCP SERVER

NO CODE

CLOUD HOSTED

HelpCrunch

HelpCrunch MCP automates your customer support workflow by connecting directly to your chat and customer records. Your agent can list conversations, update client tags, assign chats to specific team members, and pull full transaction history—all through natural conversation.

A+ Quality Score 100/100

multi-channel-support

customer-communication

chat-automation

helpdesk

customer-profiles

email-automation



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 — 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.

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.

HelpCrunch MCP

11 tools available

Cloud-hosted on Vinkius

Managing customer communication shouldn't mean jumping between five different dashboard tabs just to get one answer. This MCP lets you take control of your entire support process using only your AI client. You can ask your agent to find every chat associated with a specific customer, pulling up their full profile and interaction history in seconds. Need to hand off a complex ticket? Your agent will check the active conversations, reassign the chat to a different department, and update its status—all without you touching the HelpCrunch dashboard. Because this connection is housed within Vinkius, your AI client can access all of these support tools from one place. It functions as a dedicated Support Lead that manages everything from tagging clients to ensuring the right person gets the query.

Core Capabilities

01 — Review and search conversations

List all chats, or use filters to find specific open or historical discussions.

02 — Access detailed customer data

Pull comprehensive profiles for any customer, including their full history of interactions.

03 — Update client records and tags

Apply specific labels or tags to a customer profile after an interaction to aid future sorting.

04 — Manage team workload and routing

Change who owns a chat, assigning it immediately to the correct team member or department.

05 — Draft communications within chats

Send messages directly into an existing chat thread as part of your workflow.

One Click on Vinkius — From Prompt to Execution

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

- 01 First, subscribe to the HelpCrunch MCP on Vinkius and enter your required API key.
- 02 Next, tell your AI client exactly what you need—for example, 'Find all open chats for VIP customers in Billing.'
- 03 Your agent uses these instructions to query HelpCrunch, retrieves the necessary data (like customer details or chat transcripts), and presents the summary directly back to you.

The bottom line is that your AI client acts as an operational bridge, letting you run complex support tasks using simple commands instead of clicking through multiple web forms.

Built For

This MCP is for anyone whose job requires constant context-switching between communication platforms and internal ticketing systems. If your day involves reviewing customer history, making sure the right team member sees an issue, or updating a client's status after a call, this is built for you.

Support Manager

You use this to monitor overall department health, list departments, and check on team workloads without running separate reports.

Customer Success Representative

You rely on it to pull up a client's full history and add specific tags or updates during follow-up calls. You use `get_customer_details` frequently.

Sales Development Rep (SDR)

You monitor new inbound chats, searching through them to quickly qualify leads and ensure they are routed correctly.

11 Tools Available

#	TOOL	DESCRIPTION
01	<code>add_customer_tag</code>	Applies a specific label or tag directly to any customer's profile.
02	<code>send_message</code>	Sends a direct message into an active chat conversation for immediate communication.
03	<code>get_chat_details</code>	Retrieves all detailed information about one specific, individual chat thread.
04	<code>get_customer_details</code>	Fetches the complete profile and history for a single customer ID.
05	<code>list_chats</code>	Lists all active and historical conversations to give you an immediate overview of ongoing support work.
06	<code>list_customers</code>	Pulls a comprehensive list of every contact record stored in HelpCrunch.
07	<code>list_departments</code>	List all support departments
08	<code>list_chat_messages</code>	Useful for understanding context or historical interactions. List all messages within a specific chat
09	<code>search_chats</code>	Pass filter criteria as a JSON string in "filter_json" (e.g., {"status": "open"}). Search for chats using filters
10	<code>update_chat_assignee</code>	Assign a chat to a specific team member
11	<code>update_chat_status</code>	Update the status of a chat (e.g., closed, open)

See It in Action

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

- U** List all open chats and show the last message for each.



I've retrieved your open chats. You have 3 active conversations: 'Billing query' from Sarah, 'API Issue' from Mark, and 'Feedback' from Elena. Sarah's last message was 'When is my next invoice?'. Would you like to see the full transcript for any of these?

- U** Search for all chats from the customer with email 'john.doe@example.com'.



Searching database... I found 2 historical chats for John Doe. The first was about 'Onboarding' (Closed) and the second is an 'Upgrade Request' (Pending). Should I retrieve the messages for the pending request?

- U** Tag customer ID 5592 with 'VIP' and 'Priority Support'.



Tags added! I've successfully applied the 'VIP' and 'Priority Support' labels to customer 5592. They will now be clearly identified in your dashboard. Should I also check if they have any open chats?

Frequently Asked Questions

01 How do I find my HelpCrunch API Key?

Log in to your HelpCrunch dashboard, navigate to ****Settings > Developers > Public API****. You will be able to generate and copy your unique Bearer token from there.

02 Can I reassign a chat to another team member?

Yes! Use the ``update_chat_assignee`` tool. You must provide the ``chat_id`` and the numeric ``assignee_id`` of the team member you wish to assign the conversation to.

03 How can I filter chats by status?

Use the `search_chats` tool and provide a JSON filter string. For example, to find open chats, pass `{ "status": "open" }` in the `filter_json` parameter.

04 Is it possible to tag customers through this integration?

Yes, you can use the `add_customer_tag` tool. Provide the `customer_id` and the name of the tag you want to add to their profile.

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 `"helpcrunch": { "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

HelpCrunch 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 HelpCrunch. 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	HelpCrunch MCP
Server ID	019d75af-6a35-718c-9a07-c26842fd945a
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/helpcrunch.