MCP SERVER | NO CODE | CLOUD HOSTED

Feathery MCP

Manage form submissions and user data instantly

Feathery MCP lets you manage form submissions and user data directly through your AI agent. Instead of navigating multiple dashboards to check forms, user profiles, or API logs, you ask your agent and get instant answers about who filled out what, when, and if the connection failed.

A+ Quality Score 100/100

form-builder | user-onboarding | data-collection | workflow-orchestration | submission-tracking

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

SCAN TO VIEW

The connectivity layer between AI and the world's software.

Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the connectivity layer where AI connects 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.

Feathery MCP

11 tools available
Cloud-hosted on Vinkius

Need to understand how users interact with your application's forms? This MCP connects your Feathery account to your AI client, giving you full visibility into form automation and user data management using natural conversation. You can ask your agent to list all active forms or fetch detailed submission records for a specific person—it pulls the granular field data instantly. Need to debug why lead syncing broke? Just query the API connector logs to verify data flow errors without leaving your chat window. The Vinkius catalog makes this possible, letting you manage user profiles and workflow statuses from one place. This gives product operations teams instant visibility into every step of a user's journey.

Core Capabilities

01 — Check User Data

Retrieve all field values submitted by any specific user across their entire history with your forms.

02 — Monitor Form Progress

Get the current state of a form session to see exactly where a user left off and if they got stuck on a certain question.

03 — Audit Connections

List recent API connector logs to quickly diagnose data synchronization errors between Feathery and other services.

04 — View Form Inventory

Get a list of all active forms in your account, along with their structural details and metadata.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and input your Feathery API Key into the Vinkius platform.

02 Connect your preferred AI client (like Claude or Cursor) to the Vinkius ecosystem.

03 Ask your agent a question, such as 'Show me all active forms' or 'What data did user X submit?' and get an immediate, structured response.

The bottom line is you talk to your AI client like talking to a human teammate who has direct read access to every corner of your Feathery account.

Built For

This MCP is for the Product Operations Manager who spends too much time checking dashboards and spreadsheets. It's for the Customer Success agent who needs immediate, deep user history without escalating a ticket. If your job involves tracking data flow or troubleshooting form submissions, you need this.

Product Operations Manager

They use it to check if automated workflows are running correctly and to inspect connector logs when lead data fails to sync.

Customer Success Agent

They pull up a user's submission history quickly, allowing them to personalize support interactions based on exactly what the customer entered months ago.

Marketing Engineer

They validate lead data flow and audit API connector health directly from their chat interface, instead of logging into the backend.

What Changes When You Connect

01 Stop switching between tabs just to check form progress. You can ask your agent for `get_form_session` status, telling you exactly where a user got stuck or what they last saw.

02 Never manually audit data flow again. By using `list_connector_logs`, you get immediate error reports, letting you verify sync health and troubleshoot integration failures in seconds.

03 When a customer calls with questions about their account, use the agent to run `get_user_data` for that user ID. You instantly see every field they've filled out across all your forms.

04 You can get a full inventory of assets by running `list_forms` or `list_workflows`, giving you immediate oversight into what your company has built and how it runs.

05 Product Ops teams can use the agent to check user profiles via `list_users`, so they can track submissions and ensure compliance without leaving their main chat interface.

Real-World Applications

Debugging a failed lead sync

A Marketing Engineer notices that some leads aren't making it to the CRM. Instead of logging into Feathery, they ask their agent to run `list_connector_logs` for the relevant form. The agent finds the 'API rate limit exceeded' error immediately, allowing them to adjust the sync frequency without delay.

Handling a complex support query

A Customer Success Agent receives a request from an existing user. They ask their agent to run `get_user_data` for that specific user ID. The agent instantly returns the 'Premium Plan' selection and the full onboarding history, letting the agent solve the issue in minutes instead of hours.

Reviewing a new product launch form

A Product Operations Manager needs to ensure all forms are up-to-date. They ask their agent to run `list_forms` and review the metadata for every active form, ensuring no structural changes were missed before deployment.

Debugging a user journey stall

A team member notices that new signups are abandoning the onboarding process. They ask their agent to check `get_form_session` for several users and discover that every stalled session is failing at Step 3, pointing directly to the friction point.

Patterns to Avoid

Checking logs manually

✗ AVOID

Logging into Feathery, finding the 'Connectors' tab, and then scrolling through pages of timestamps looking for a specific error code or user ID.

✓ INSTEAD

Ask your agent to run `list_connector_logs` directly. The agent filters and surfaces only the errors you need, giving you the exact details without manual searching.

Gathering user data piece by piece

✗ AVOID

Having to open a user profile, then opening the submission history, then manually checking which forms were used just to compile one report.

✓ INSTEAD

Use `get_user_data` for that user. The agent aggregates all field values from every form they submitted into one clean output.

Guessing workflow status

✗ AVOID

Assuming a complex automated process is working because the last step ran, only to find out an earlier required step failed silently.

✓ INSTEAD

Run `list_workflows` and then `get_workflow_details`. This confirms the execution status of every single component in your automation.

The Right Fit

Use this MCP if your primary pain point is data visibility. Specifically, you need to connect form submissions, user profiles, or complex workflow logs into a conversational interface. If your job involves asking questions like 'What did they submit?' or 'Why did the sync fail?', this is for you. Don't use this if you just need to *build* forms—that's Feathery itself. Also, don't use it if you only need simple user lists; while `list_users` works, other dedicated CRM tools might give you better relational data views. This MCP shines when you need the raw operational details that live inside your form builder.

The Daily Struggle with Data Silos

Right now, if a customer calls about an onboarding issue, you have to jump through three different systems. You check the user's profile in one tab, then go to another dashboard to see which forms they used, and finally open a third panel just to manually check if the data synced correctly to the CRM. It takes five clicks, two minutes, and three copy-pastes.

With this MCP, you ask your agent: 'Check user X's form submissions.' The agent aggregates all that information—the submission history, the workflow status, and even connector logs—and gives it back in a single readable response. It's instant access to everything.

Get Full Form Control with Feathery MCP

Manual form auditing means you're constantly running `list_forms` and then clicking into each one just to verify the metadata or check if a workflow is attached. It's tedious, error-prone work that slows down product launches.

Now, you simply ask your agent to run `get_form_details`. You get all the structural information instantly. This means your team can confidently audit and deploy new forms without spending hours clicking through admin menus.

Feathery - 11 Tools

These tools let you programmatically interact with all aspects of your Feathery account, from listing users to fetching detailed form submission records.

#	TOOL	DESCRIPTION
01	`get_workflow_details`	Gets detailed information about a single, automated workflow process.
02	`list_connector_logs`	Lists recent API connector error logs for troubleshooting data synchronization issues on a specific form.
03	`list_environments`	Shows all available Feathery development or staging environments you have configured.
04	`list_forms`	Lists every active and inactive form within your entire Feathery account.
05	`get_account_info`	Retrieves core account details for your Feathery setup.
06	`get_form_details`	Gets the structure and metadata for a single, specified form.
07	`get_form_session`	Checks the current progress or session state of a specific form for any user.
08	`get_me`	Verifies the identity and details associated with your API token connection.
09	`get_user_data`	Retrieves all field values submitted by a specific user across every form they've interacted with.
10	`list_users`	Retrieves a list of all user accounts connected to your Feathery environment.
11	`list_workflows`	Lists all automated workflows that are currently set up in your account.

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 forms in my account.

I've retrieved your forms. You have 'User Onboarding' (ID: form_123), 'Customer Feedback' (ID: form_456), and 'Lead Gen' (ID: form_789) currently active. Which one should I inspect?

U Show me the data submitted by user user_99.

Fetching data for user_99... They submitted 'John Doe', 'john@example.com', and selected 'Premium Plan' in the onboarding form. Would you like to see their full session history?

U Check if there are any connector errors for the Onboarding form.

Inspecting connector logs... I found one error for form_123: 'Klaviyo API rejected email format'. All other syncs were successful. Shall I retrieve the user ID associated with this error?

Frequently Asked Questions

01 **How do I use Feathery MCP to check user data?**

You ask your agent to run `get_user_data` for the specific user ID. The agent then compiles all field values submitted by that user across every form they've ever completed.

02 **What is the best way to check API errors using Feathery MCP?**

You should run `list_connector_logs`. This tool retrieves recent error logs for a specific form, letting you diagnose sync failures without having to navigate the backend settings.

03 **Can I see all my active forms with Feathery MCP?**

Yes. Use the `list_forms` tool. It returns a complete list of every form in your account, giving you an immediate inventory of all assets.

04 **Does Feathery MCP help me see where a user left off on a form?**

Absolutely. Running `get_form_session` allows the agent to retrieve the current state for any specific form, showing exactly which question needs attention or if they abandoned the process.

05 **How many tools are available in Feathery MCP?**

This MCP provides access to 11 distinct operational tools that cover everything from listing user accounts to checking API connection health.

06 **How do I obtain my Feathery API Key?**

Log in to your Feathery dashboard, navigate to **Settings > API**, and generate a new key. Ensure you copy the token immediately as it is only shown once.

07 **Can I see the data submitted by a specific user?**

Yes! Use the `get_user_data` tool with the specific User ID to retrieve all field values submitted by that user across any of your forms.

08 **How can I troubleshoot integration errors via chat?**

The `list_connector_logs` tool retrieves recent error logs for your API connectors. This allows your agent to identify exactly which field mapping or endpoint caused a failure.

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 `"feathery": { "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

Feathery 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 Feathery. 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	Feathery MCP
Server ID	`019d7597-c120-73e8-8ebc-47bd27e1934e`
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/feathery.