

MCP SERVER

NO CODE

CLOUD HOSTED

Gravity Forms MCP

Manage Every Form Submission, Automatically.

Gravity Forms MCP gives your AI agent full control over WordPress form submissions. You can list all forms on a site, pull specific entry data, update old records, or even programmatically submit new leads—all through natural conversation with your preferred client.

A+ Quality Score 98.33/100

form-builder

lead-capture

data-entry

wordpress-plugin

submission-management

rest-api



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.

Gravity Forms MCP

12 tools available

Cloud-hosted on Vinkius

Connect this MCP to any compatible AI client and take charge of your entire Gravity Forms setup. Your agent can now manage everything related to form entries and submissions without you having to log into the WordPress admin dashboard. Want to know what forms exist? You can list them all and inspect their field structures. Need to track a specific customer? Retrieve detailed submission records, or update an old entry's values if the client changed their mind. If your team needs context on a tough case, you can read and even add internal notes directly to the submission record. It also lets you look at aggregate results for any form, so you can analyze conversion rates right in the chat window. We built this integration into Vinkius so you connect once and get access to all your data management tools.

Core Capabilities

01 — List available forms

You retrieve a list of every form created on the site, including their IDs.

03 — Retrieve all submissions

You pull a list of submitted entries across multiple forms, allowing you to search by date or criteria.

05 — Update existing entries

You modify specific fields in a submitted form record, correcting errors or adding missing information.

07 — Analyze form performance metrics

You pull aggregate statistics and results to track how well a particular form is performing over time.

02 — Get form structure and settings

The agent reads the specific fields and configuration details for any given form ID.

04 — Get specific entry data

The agent fetches the full details and field values for one specific form submission.

06 — Add internal notes to submissions

The agent attaches private, visible context notes to a submission for the support team's records.

One Click on Vinkius — From Prompt to Execution

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

- 01** First, subscribe to this MCP on Vinkius and provide your Site Base URL, Consumer Key, and Consumer Secret from your WordPress Gravity Forms settings.
- 02** Next, tell your AI client exactly what you need. For example: 'List all forms and then check the submissions for Form ID 5.'
- 03** Finally, the agent executes the required actions—retrieving data, updating records, or calculating statistics—and presents the results back to you directly in the chat.

The bottom line is that your AI client handles all API calls and data retrieval, letting you manage forms without touching a browser.

Built For

Site Administrators who spend too much time manually checking form setups or Support Agents drowning in ticket histories. If tracking down customer submissions or updating internal records is tedious clicking, this MCP saves your afternoon.

Support Team Manager

The manager uses this to quickly look up specific customer entries and add private notes, keeping resolution history centralized without digging through tickets.

Digital Marketing Specialist

This person uses it to analyze form results across different campaigns, checking conversion rates and identifying which forms need better placement on the site.

Web Administrator

The admin tests new form layouts by listing all forms and retrieving their field structures before going live, ensuring everything works correctly.

What Changes When You Connect

- 01 Instead of logging into WordPress to check submissions, your agent handles it. You can instantly retrieve specific entry data using the `get_entry` tool and analyze who submitted what without manual searching.
- 02 Support teams save hours by centralizing context. Use the `create_entry_note` tool to add internal notes right alongside the customer's submission, keeping a perfect record for faster resolution.
- 03 Marketing analysis becomes instant. With one call to `get_form_results`, you pull aggregate statistics and conversion data—no need to export CSVs or build custom reports in Google Sheets.
- 04 You can test submissions safely. Use the `submit_form` tool to programmatically trigger a form submission, running validation checks without needing to physically fill out a browser form first.
- 05 System auditing is simple. The agent lets you list all forms (`list_forms`) and check their configuration (`get_form`), ensuring no critical fields or settings were accidentally changed.

Real-World Applications

A customer calls with a billing question, but the original form entry is old.

Instead of making the agent ask for an ID and then searching by date, you simply tell it: 'Find me the submission for Jane Doe from last month.' The agent uses `list_entries` and `get_entry` to pull up the exact data needed immediately so the support team can process the refund.

The marketing team needs to know which service request type is failing most often.

You ask your agent to analyze performance. It uses `get_form_results` on the 'Service Request' form, quickly identifying that 70% of submissions fail validation because a required field was missed—a stat you wouldn't get easily otherwise.

The web admin needs to confirm if a third-party CRM is receiving new leads.

You instruct your agent: 'Check the feeds for the Contact Us form.' It uses `list_feeds` and confirms that the connection is active, validating data flow before launching a major campaign.

A bug report comes in saying a submission was lost.

You ask your agent to confirm all submissions for Form ID 3 from last Tuesday. The agent uses `list_entries` and finds the missing record, confirming it was simply buried under thousands of other entries.

Patterns to Avoid

Treating form data like a simple database table

X AVOID

A user might try to 'get all fields for all forms' in one go, expecting a clean list. This fails because the structure and configuration must be retrieved per form ID.

✓ INSTEAD

Always start by using `list_forms` to get the IDs you need, then iterate through them calling `get_form` for each individual ID. Never assume global knowledge.

Trying to update a submission without knowing its ID

X AVOID

A user might say: 'Change John Doe's email.' The agent can't act because it needs the unique entry identifier, not just the name.

✓ INSTEAD

First, use `list_entries` or `get_entry` to find and confirm the specific Entry ID. Once you have that number, then use `update_entry`.

Confusing submission data with internal notes

X AVOID

A user might try to delete a note they think is part of the record. This fails because notes are separate metadata attached using `create_entry_note`, not core entry fields.

✓ INSTEAD

If you want to manage context, always use dedicated tools like `list_entry_notes` or `get_entry` first. Never assume a note can be edited as if it were submission data.

The Right Fit

Use this MCP if your primary need is managing the lifecycle of submitted form data—from initial capture to final analysis, and tracking internal context. You should use this when you need to perform actions like updating specific fields (`update_entry`), or running aggregate reports on submission types (`get_form_results`). Don't use it if you just need general WordPress content management (like editing posts), because this tool is strictly

focused on the form data layer. If your goal is simply reading a blog post, connect a different MCP. This is for deep data interaction with Gravity Forms.

The Headache of Manual Form Audits

Right now, checking submissions means bouncing between tabs: the main entry list, the individual submission details, and then opening a separate internal note section just to see if someone already looked at it. You spend minutes clicking through IDs, copy-pasting data points, and cross-referencing fields just to get one complete picture of what happened.

With this MCP, that entire manual process collapses into conversation. Your AI agent handles the clicks and the context switching. You ask for a full status report on an entry, and it pulls together the submission details, checks if notes exist, and gives you everything in one response.

Get Full Control Over Form Data with `get_form`

Previously, figuring out what data a form was collecting required logging into the editor and clicking through complex field groups. You had to manually verify if every single field—like 'Phone Number' or 'Preferred Service'—was configured correctly before trusting it.

Now, your agent uses `get_form` to instantly pull the entire structure of the form. It presents you with a clean breakdown of fields and settings. That's how accurate and fast your data validation becomes.

Gravity Forms: 12 Tools for Data Management

These tools allow your AI client to interact with every part of the Gravity Forms API. You can read form structures, retrieve specific submissions, update records, and analyze performance metrics.

| # | TOOL | DESCRIPTION |
|----|--------------------------------|---|
| 01 | <code>create_entry_note</code> | Attaches a private note to a specific submitted form entry for internal team context. |
| 02 | <code>delete_entry</code> | Permanently removes an entire form submission record from the site database. |
| 03 | <code>get_entry</code> | Fetches all detailed information and field values for one specific form entry. |
| 04 | <code>get_form_results</code> | Retrieves aggregate statistics, helping you analyze the overall performance of a particular form. |
| 05 | <code>get_form</code> | Provides the complete structure and configuration details for a specific form ID. |
| 06 | <code>get_notifications</code> | Checks which external services are receiving notifications when a form is submitted. |
| 07 | <code>list_entries</code> | Pulls a list of all submissions across the site, allowing you to narrow down results by date or criteria. |
| 08 | <code>list_entry_notes</code> | Retrieves all internal notes that have been attached to a specific form entry ID. |
| 09 | <code>list_feeds</code> | Lists the active data feeds connected to a specific form, ensuring external services are getting data. |
| 10 | <code>list_forms</code> | Retrieves a list of every single form configured within your Gravity Forms site. |
| 11 | <code>submit_form</code> | Submits new data to a specific form programmatically by passing in the required field values. |
| 12 | <code>update_entry</code> | Modifies an existing, saved form entry record with new data. |

See It in Action

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

U List my forms in Gravity Forms



Retrieving your forms... I found 3 active forms: 'Contact Us' (ID: 1), 'Service Request' (ID: 2), and 'Customer Feedback' (ID: 3). Would you like to see the entries for the service request form?

U Show me the entries for Form ID 1 from this week



Fetching entries... I found 5 new submissions this week for the 'Contact Us' form. The latest is from John Doe (Entry #1042). Would you like to read the details of his message?

U Add a note to entry #1042: 'Customer contacted support, refund processed'



Adding note... Success! I've added the internal note to Entry #1042. It is now visible to the support team in your WordPress dashboard.

Frequently Asked Questions

01 How do I list all my forms using Gravity Forms MCP?

You use the `list_forms` tool to get a complete catalog of every form on your site. This immediately gives you all the Form IDs necessary to work with them.

02 Can I update an entry's data using Gravity Forms MCP?

Yes, use `update_entry`. You need two things: the unique Entry ID of the record and the specific field values you want to change. This prevents accidental modification.

03 Does Gravity Forms MCP only handle new submissions?

No, it handles the entire lifecycle. You can list old entries using ``list_entries``, retrieve details with ``get_entry``, and even delete records entirely using ``delete_entry``.

04 What is the difference between getting an entry and listing submissions?

Using ``list_entries`` gives you a high-level summary of many entries, like a quick table view. Using ``get_entry`` drills down to show every single field value for one specific submission.

05 How do I document context for the support team using Gravity Forms MCP?







You use the ``create_entry_note`` tool. This attaches a private, internal note that is visible to your team but won't appear in the submission data itself.

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>"gravity-forms": { "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

Gravity Forms 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 Gravity Forms. 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 | Gravity Forms MCP |
| Server ID | 019d75aa-f6ab-7020-931f-f4f710639e7d |
| 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/gravity-forms.