

MCP SERVER

NO CODE

CLOUD HOSTED

Memberful MCP

Manage your membership billing and member data instantly.

Memberful MCP connects your AI agent directly to your subscription platform data. It lets you manage membership records, monitor active subscriptions, and review sales orders—all through natural conversation. Stop logging into dashboards; get real-time member metadata, track recurring revenue health, and oversee financial details from one single chat window.

A+ Quality Score 100/100

membership-management

recurring-revenue

gated-content

subscription-billing

member-analytics

monetization



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.

Memberful MCP

12 tools available

Cloud-hosted on Vinkius

Memberful gives your AI client direct access to the core functions of your paid membership site. Instead of manually navigating complex admin panels, you simply ask your agent what you need to know about your audience or sales pipeline. You can check who is a member and pull their exact profile details. Need to see if someone's account has lapsed? Your agent monitors active subscriptions and expiration dates instantly. Over time, this makes managing recurring revenue simple. Furthermore, the MCP allows you to review past transactions by listing recent orders or even pulling master data for available plans and discounts. It turns complex membership billing into a conversation. By connecting your agent through Vinkius, you centralize all that critical member lifecycle information without ever leaving your chat interface.

Core Capabilities

01 — Audit Member Records

List all current members and retrieve specific profile data for any individual account.

02 — Track Billing Status

Monitor active subscriptions, find expiration dates, and check assigned plans to assess revenue health.

03 — Review Financial History

List recent sales orders or retrieve detailed transaction metadata for financial oversight.

04 — Manage Promotional Assets

Get master data about current membership plans and available discount codes.

05 — Handle New Accounts

Create new member profiles directly through the chat interface.

One Click on Vinkius — From Prompt to Execution

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

- 01** Subscribe to this MCP and enter your Memberful API Key and Site Name (subdomain) into Vinkius.
- 02** Connect your agent to the Vinkius catalog. This gives your AI client permission to access your membership data streams.
- 03** Ask your agent a question, like 'Show me all annual subscribers whose plans expire next month.' The MCP runs the query and delivers the answer.

The bottom line is you get an automated operations assistant that speaks the language of your billing platform, without needing to know any API calls.

Built For

This MCP is for founders and managers who spend too much time switching between their membership dashboard, spreadsheet, and email. If you struggle to get a quick snapshot of your revenue health or need to check member status without logging in, this connector saves hours.

Content Creator / Publisher

Uses the MCP to quickly list members and monitor churn rates when they need immediate data for marketing copy.

Operations Manager

Automates the retrieval of subscription metadata, checking plan distribution and managing webhook details via natural conversation.

Billing Administrator

Checks recent sales orders and lists all active coupons to ensure pricing and payments are accurate for new customers.

What Changes When You Connect

-
- 01 Never manually check the status of a subscriber again. You can use `list_subscriptions` to monitor expiration dates, allowing you to proactively flag accounts needing renewal before they lapse.

 - 02 Getting an overview of sales is simple. Instead of digging through logs, your agent uses `list_orders` to give you a summary of recent transactions and total revenue for the day.

 - 03 Managing pricing details used to mean checking multiple dashboards. Now, simply ask for available plans using `list_plans`, or check promotional options with `list_discounts` in one query.

 - 04 When onboarding new paying members, your agent handles the whole process. You can use `create_member` and immediately follow up by retrieving their metadata using `get_member` to confirm setup.

 - 05 The MCP centralizes operational visibility. Instead of having separate tools for member lists and financial records, you get one chat interface that provides both data streams.
-

Real-World Applications

Checking a client's membership status before a call

A sales manager needs to know if a potential high-value client is currently subscribed. They ask their agent, 'What is the status of John Doe?' The agent runs `get_member` and reports back that he is on the 'Annual Supporter' plan, expiring in 90 days. This lets the manager tailor the conversation immediately.

Auditing recent revenue spikes

The finance team needs to reconcile a large batch of sales and must verify if discounts were applied correctly. They ask the agent to 'List all orders over \$50 from last week.' The MCP uses `list_orders` and provides transaction summaries, allowing for quick discrepancy checks.

Onboarding a new major corporate partner

An ops manager needs to create 20 accounts manually. Instead, they ask the agent to 'Create twenty members with standard access.' The MCP uses ``create_member`` repeatedly and confirms each user account is active.

Understanding marketing campaign success

A marketing lead ran a discount campaign and needs to know which plans were popular. They ask the agent, 'What are the top three most listed plans?' The MCP uses ``list_plans`` and provides the master data required for immediate analysis.

Patterns to Avoid

Checking member status in multiple tabs

X AVOID

A user has to navigate from the 'Members' tab, then click into a specific profile, and finally cross-reference that data with the separate 'Subscriptions' ledger just to find an expiration date.

✓ INSTEAD

Instead, ask your agent directly. Your agent combines the necessary information by calling ``get_member`` and automatically checks the subscription status for you. It delivers the final answer without showing multiple tabs.

Manually updating discounts after a sale

X AVOID

A sales rep makes a deal but forgets to apply the correct promotional code, forcing an ops worker to manually go into the billing backend and change pricing records.

✓ INSTEAD

The agent keeps track of available codes. You can use ``list_coupons`` or ``list_discounts`` first to confirm the correct promotion, then ask the agent to record it when creating a new account.

Guessing if an account was paid for

X AVOID

A team member sees a member listed but can't tell if they are active or just on a free trial because they have to check multiple internal status flags.

✓ INSTEAD

Ask the agent to 'List all subscriptions.' This tool uses ``list_subscriptions`` and gives you the immediate, authoritative billing status for every account.

The Right Fit

Use this MCP if your primary pain point is data retrieval related to membership, subscriptions, or revenue. If you need to know 'Who is paying? What plan do they have? When does it expire?' then this is your tool. It excels at combining member identity (`get_member`) with billing status (`list_subscriptions`). Don't use this if you primarily need to manage content permissions (e.g., marking a document as

private). For that, you might need a dedicated file access or CMS connector. Also, don't rely on it for deep accounting ledger work; stick to using `List_orders` for summarizing financial history only.

Tracking revenue requires logging into five different dashboards.

Every time you need a simple status update—like checking if a member's annual fee was paid, or seeing which discounts are available for new sign-ups—you spend minutes clicking through Memberful's dashboard. You switch from the membership list to the billing section, then maybe over to the plans page, just to piece together one answer.

With this MCP, you stop clicking and start talking. You ask your agent a complex question about member status or order history, and it pulls all that data for you in one response. The information is immediately available without opening a single new tab.

Memberful makes account creation instant with ``create_member``.

Previously, creating a member meant filling out multiple forms: one for the basic profile data, another to set up the initial plan, and maybe a third to generate an internal tracking code. It was slow, error-prone work that required handoffs between departments.

Now, you tell your agent what needs to happen —'Create a new member with premium access.' The MCP uses `create_member` to handle it all in one go. You get the confirmation immediately; there are no manual steps left.

Memberful: 12 Tools for Membership Management

These tools let you perform every core function of member management—from creating accounts to checking subscription status—using natural language through your AI agent.

#	TOOL	DESCRIPTION
01	<code>create_member</code>	Adds a brand-new user account to your member directory.
02	<code>get_account_info</code>	Retrieves general site information about the membership platform itself.
03	<code>get_member</code>	Pulls all specific details for a single member ID or email address.
04	<code>get_order</code>	Grabs the full details of one specific transaction or purchase.
05	<code>list_coupons</code>	Shows all coupon codes that are currently active and available to members.
06	<code>list_discounts</code>	Lists every discount code or rule set applied to member pricing.
07	<code>list_downloads</code>	Retrieves a list of downloadable assets associated with members.
08	<code>list_members</code>	Gets a complete roster or summary of every member currently on the platform.
09	<code>list_orders</code>	Pulls a list summarizing recent customer purchases and transactions.
10	<code>list_plans</code>	Retrieves master data for all available membership subscription plans.
11	<code>list_subscriptions</code>	Lists every active or expired subscription account and its status.
12	<code>list_webhooks</code>	Shows the system's configured webhooks to monitor automated workflow connections.

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 membership plans in Memberful.



I've retrieved your plans. You have 3 active options: 'Premium Monthly', 'Annual Supporter', and 'Founding Member'. Which one would you like to see subscriber counts for?

U Show the last 5 orders in my Memberful site.



I've retrieved your recent orders. You had 5 transactions today totaling \$450, including new subscriptions for 'Alice Miller' and 'Robert Chen'.

U Get details for member ID '12345'.



I've retrieved the metadata for member ID 12345. It belongs to Sarah Jones (sarah@example.com), who is currently on the 'Annual Supporter' plan expiring in 6 months.

Frequently Asked Questions

01 How do I check a member's details using Memberful MCP?

You use the `get_member` tool to pull all metadata for any single user. This gives you their profile information, current plan status, and other associated data points.

02 Can I list all my active subscriptions with Memberful MCP?

Yes, the `list_subscriptions` tool retrieves a complete list of every subscription account. This is crucial for running reports on your overall revenue health.

03 Does Memberful MCP help me with pricing? Is there a plans tool?

Absolutely. Use `list_plans` to get master data on all active membership options, or use `list_discounts` to see what promotional codes are currently valid.

04 What if I need to add a new member account through Memberful MCP?

You run the `create_member` tool. This sends the necessary data to your platform, instantly adding the user and initiating their membership process.

05 Is this only for active members? Can I check past orders?







No. You can use `list_orders` to see a history of transactions regardless of whether the member is currently subscribed or not. This helps with auditing and reporting.

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>"memberful": { "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

Memberful 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 Memberful. 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	Memberful MCP
Server ID	019dd122-806c-72a1-ab01-308e8bbf3ce3
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/memberful.