MCP SERVER | NO CODE | CLOUD HOSTED

Medium MCP

Publish stories and manage your entire writing ecosystem.

Medium MCP lets your AI agent manage your entire publishing workflow. You can publish stories directly, audit all associated publications, and query contributor lists—all without opening a dashboard. It treats content distribution like a natural conversation with your writing team.

A+ Quality Score 100/100

blogging | publishing | content-creation | writing | audience-engagement | social-platform

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/medium-alternative

SCAN TO VIEW

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.

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.

Medium MCP

5 tools available
Cloud-hosted on Vinkius

This connector turns Medium into something far more than just a blogging platform; it makes it an operational hub for writers and editors. By connecting this MCP to Vinkius, you give your agent the ability to handle publishing tasks that used to require multiple manual logins and complex forms. Your agent can instantly list every publication you own or contribute to, grab detailed user information, and create entirely new posts tailored for specific audiences. When you're working on a high-traffic publication, this MCP acts like a real-time editorial assistant. It handles the technical formatting—whether you need Markdown or HTML—and ensures that content gets distributed exactly where it needs to go, whether that's your profile or a dedicated section of a larger journal.

Core Capabilities

01 — Audit all publications

Retrieve and list every publication associated with your Medium account.

02 — Check writing team membership

Query any specific publication to get a detailed roster of its contributors.

03 — Create standalone stories

Draft and publish new articles directly to your main profile feed.

04 — Targeted content publishing

Publish a story specifically under one of your existing publications, ensuring it reaches the right audience.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your Medium Integration Token.

02 Connect it via your preferred AI client (Claude, Cursor, or any compatible agent).

03 Ask your agent to perform a publishing task, like listing publications or creating a post.

The bottom line is that once connected, you just talk to your agent; it handles the Medium details for you.

Built For

This MCP is built for content teams and solo writers who get frustrated by manual publishing checklists. It's perfect for the editor tired of cross-referencing contributor lists, or the marketing lead needing to rapidly audit tags across multiple published pieces.

Content Creator

Uses this MCP to publish drafts and stories directly from their writing workflow without switching tabs.

Editorial Manager

Audits publication statuses and verifies the full contributor list for a journal before a major content drop.

Technical Writer

Creates new posts, ensuring the content is formatted correctly in both Markdown and HTML styles.

What Changes When You Connect

01 You can instantly list all publications using the `list_publications` tool. You never have to remember which journal you contribute to; the agent handles that inventory check for you.

02 When you write a story, you don't just publish it. You use `create_publication_post` to ensure the article lands exactly where your audience expects it, under the right publication banner.

03 Need to verify who wrote what? The `list_contributors` tool gives you a clean roster of writers for any journal, so you always know who's on the team page.

04 Forget formatting headaches. You can publish content using either HTML or Markdown formats, ensuring your aesthetic stays perfect regardless of how it gets exported.

05 Start by calling `get_me` to pull your user data. This gives your agent a solid foundation of context before it starts drafting and publishing anything.

Real-World Applications

Updating the 'About Us' page for a new journal.

The editorial manager needs to verify all current writers. They ask their agent to run `list_contributors` for the main publication ID, quickly confirming that no team members were accidentally left off the roster.

Batch publishing evergreen content.

A marketing team has 10 finished articles and needs them published immediately. They use `create_publication_post` in a loop, ensuring each piece lands under the correct publication with custom tags.

Onboarding a new writer to the platform.

The content creator wants to show their agent all available publishing outlets. They call `list_publications`, getting a clean list they can use to write onboarding instructions for a new hire.

Patterns to Avoid

Trying to publish everything manually.

✗ AVOID

Copying the same article text, changing the title, and hitting 'Publish' in multiple browser tabs for every publication you belong to. This takes forever.

✓ INSTEAD

Let your agent use `create_publication_post`. You give it the content and the target publication name once, and it handles the distribution across all necessary locations.

Forgetting who owns which piece of content.

✗ AVOID

When reviewing a journal's contributors, you have to manually scroll through profiles to confirm everyone listed is actually active or belongs there anymore.

✓ INSTEAD

Use the `list_contributors` tool. It immediately provides an accurate list of members for that specific publication, saving tedious manual verification.

Drafting content without knowing your profile status.

✗ AVOID

Starting to write a huge article but forgetting if you need to update your personal bio or check which publications are even active under your account.

✓ INSTEAD

First, run `get_me` to pull the latest user data. Then, use `list_publications` to confirm all available destinations for your new content.

The Right Fit

Use this MCP if your primary bottleneck is the sheer volume and variety of publishing targets. If you need an AI agent to act like a digital assistant that knows every journal you contribute to, or needs to manage contributor lists across multiple publications, this is it. Don't use it if you only want basic writing assistance; for that, any general LLM will do fine. However, if your goal is just simple content generation without publishing oversight, then an advanced document drafting tool might be better. Remember, the power here is in orchestration: using tools like `list_publications` and `create_publication_post` together means you're automating a cross-platform editorial process that few other single connectors can touch.

The Chore of Content Distribution

Every writer knows the drill. You finish a great article, but before you hit send, you have to stop and think: 'Where does this go?' Then you open your browser tabs—one for your main profile, one for Publication A, one for Journal B. You copy the text, paste it into three different interfaces, make sure the tags are right in each place, and manually check that the formatting didn't break anywhere.

With this MCP connected through Vinkius, you ditch the tabs. Your agent receives your finished content once. It then knows to perform a targeted publishing action, using tools like `create_publication_post` to distribute it accurately across every required destination in one conversation. You just get the published story; nothing else.

Get Full Control With the Medium MCP

You eliminate manual audits. Instead of clicking through menus to see who wrote for Publication X, you simply ask your agent to run `list_contributors` and get a clean list instantly. You also don't have to guess what content formats are best; the MCP handles publishing in both HTML and Markdown.

What changes now is that your writing workflow becomes linear. You write it once, and the system ensures it gets properly published everywhere without you touching a dashboard or worrying about copy-pasting errors.

Medium Publishing Tools (5)

These tools let you programmatically interact with Medium's core functions, from publishing new content to auditing entire publication rosters.

#	TOOL	DESCRIPTION
01	`create_post`	Publishes a brand-new article directly to your Medium profile feed.
02	`create_publication_post`	Creates and publishes an article targeted under a specific, named publication.
03	`get_me`	Retrieves your current user details and profile information from Medium.
04	`list_contributors`	Fetches the list of all people who contribute to a specific publication.
05	`list_publications`	Generates a complete list of every publication you are associated with on Medium.

See It in Action

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

U Get my Medium user ID and publications.

I've retrieved your profile. Your user ID is '12345' and you contribute to 2 publications: 'The Startup' and 'AI Trends'. Which one would you like to explore?

U Publish a draft titled 'My AI Story' with tags 'AI,Future'.

Draft created! 'My AI Story' is now saved in your Medium account with 2 tags. You can review and publish it from your dashboard whenever you're ready.

U Who are the contributors for publication ID xxxx?

I've found 3 contributors for that publication: 'Jane Doe', 'Marcus R.', and 'BotAssistant'. Would you like to see their profile URLs?

Frequently Asked Questions

01 **How do I list all my publications using the Medium MCP?**

You run the `list_publications` tool. This command retrieves a comprehensive list of every publication you are associated with on your account, giving you immediate visibility into all your content destinations.

02 **Can I publish a story to only one specific journal?**

Yes, use the `create_publication_post` tool. This allows you to target your new article specifically under a named publication, ensuring it doesn't get lost on your main profile feed.

03 **What is the best way to check my user data?**

You should use `get_me`. Running this tool retrieves all necessary authenticated user details and confirms your current status, giving your agent a solid foundation for any publishing task.

04 **Does Medium MCP help me manage contributor lists?**

Absolutely. You can run the `list_contributors` tool to query any publication and get a clear roster of all contributors associated with it, making team management simple.

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 `"medium-alternative": { "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

Medium 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 Medium. 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	Medium MCP
Server ID	`019d8456-2ee2-73ab-ab73-80e4ca0aabcd`
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/medium-alternative.