MCP SERVER | NO CODE | CLOUD HOSTED

Ghost MCP

Manage posts, authors, and pages with AI.

Ghost MCP connects your AI agent directly to your Ghost CMS, letting you manage and retrieve content without logging into a dashboard. List posts, check authors, pull taxonomy tags, or view site settings—all through natural conversation. Stop copy-pasting from admin panels; start working with live publication data instantly.

A+ Quality Score 100/100

blogging | publishing | content-creation | newsletter-platform | web-publishing

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

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.

Ghost MCP

11 tools available
Cloud-hosted on Vinkius

This connection lets your AI agent read and write to your Ghost CMS directly. You can ask it to list all published blog posts, pulling the full HTML content and metadata for each one. Need to know what categories exist? Just ask it to retrieve all available taxonomy tags. It also helps you track who's writing by fetching profile details for active authors. Even site-level data is accessible; you can get global settings or list current membership tiers to understand your paywall rules. By using this MCP on Vinkius, your AI client keeps the entire publishing stack live and available without requiring you to switch apps or navigate complex menus.

Core Capabilities

01 — Manage published articles

Fetch details for specific posts by their slug, or list every article currently marked as published.

02 — Understand content structure

List all available categorization tags and retrieve global site settings to understand how your content is organized.

03 — Track contributors

Pull author profile details and list active writers across the entire publication.

04 — Review static pages

Retrieve the full content and metadata for non-blog pages using their slugs.

05 — Inspect membership data

List all active subscription tiers, helping you understand your paywall structure.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP on the Vinkius Marketplace and provide your Ghost Base URL and Content API Key.

02 Connect your preferred AI client (like Cursor or Claude) to the Vinkius catalog, granting access to the Ghost tools.

03 Start by asking a natural language question—for example, 'List all authors and their profiles'—and watch your agent execute the required tool calls.

The bottom line is that you treat your entire CMS like one accessible data source within your AI workflow.

Built For

Any content professional who spends too much time clicking through Ghost's administrative dashboard to gather simple pieces of information. This helps editorial leads and developers get critical site data instantly, right where they are working.

Content Manager

You need to pull a list of all published articles or check if an author's profile is up-to-date without leaving your writing environment.

Frontend Developer

You have to automate the retrieval of taxonomy tags and site configuration data to build new features into a client application.

Editorial Lead

You need an instant overview of all static pages and active subscription tiers to plan marketing copy or check revenue streams.

What Changes When You Connect

01 Access article content instantly: Instead of visiting the post in your browser to copy text, you can ask your agent to get post details by slug. This is faster and keeps your workflow contained.

02 Build a full site map programmatically: You don't have to click through multiple sections. Use list_static_pages and list_content_tags together to build an inventory of everything on the site via simple AI commands.

03 Understand your audience revenue: List subscription tiers instantly tells you what paywall rules are in place, letting you draft content that matches your pricing structure without manual checks.

04 Keep track of contributors easily: Use list_blog_authors and get_author_details to generate an up-to-date roster of writers and their professional bios for a newsletter sendout.

05 Get global site context on demand: Need to know how the site handles routing or what its primary title schema is? The get_site_settings tool gives you that data instantly, allowing developers to build against known standards.

Real-World Applications

Need a quick content audit for SEO

A developer needs to verify if all static pages have proper metadata. They instruct their agent to list_static_pages, then loop through the results, calling get_page_by_slug on each one to pull the full text and check for missing H1 tags.

Drafting a contributor spotlight

The Content Manager wants to write an announcement about a new writer. They ask their agent to list_blog_authors, identify the target author, and then use get_author_details to pull the necessary bio text for immediate inclusion in the draft.

Planning a premium feature rollout

The Editorial Lead needs to know how many paid tiers exist. They prompt their agent to list_subscription_tiers, allowing them to confirm the current pricing structure before announcing changes on social media.

Generating an index of all topics

A user wants a master list of everything published. They ask for list_published_posts and list_content_tags, getting two clean lists that they can then combine to create an exhaustive site index.

Patterns to Avoid

Manually copying post data

✗ AVOID

Opening the Ghost dashboard, finding a specific article, and manually copying its content or metadata into a spreadsheet for analysis. This is slow and prone to formatting errors.

✓ INSTEAD

Ask your agent to get_post_by_slug using the article's slug. The tool pulls all clean, structured data directly from the source into your AI client.

Guessing site capabilities

✗ AVOID

Assuming that a certain feature, like global routing rules or subscription limits, exists because it seems logical, but failing to confirm its actual implementation.

✓ INSTEAD

Use get_site_settings. This tool provides the authoritative data on how your entire publication is configured.

Treating content tags as a single list

✗ AVOID

Trying to figure out if 'tech' and 'development' are separate categories or just different ways of tagging the same thing, requiring multiple clicks through the dashboard.

✓ INSTEAD

First, run list_content_tags to see all options. Then use get_tag_details on a specific tag name to understand its exact metadata.

The Right Fit

Use this MCP if your core workflow revolves around reading, listing, or pulling structured data from published content, authors, and site settings. Specifically, if you need to know the full list of available taxonomy tags (list_content_tags) or verify post details by slug (get_post_by_slug), this is what you want.

Don't use it if your main goal is writing new content drafts from scratch, or managing user accounts that aren't related to authorship. If you are only trying to generate text and don't need access to existing publication data, a general-purpose LLM will suffice. Also, if you just want a list of all authors without their profile details, the list_blog_authors tool is sufficient; you won't need get_author_details.

Content publishing relies on endless clicks and copy/paste.

Today, gathering basic data means logging into Ghost, navigating through the posts section to list published articles. Then you jump over to the 'Authors' tab to find contributor bios. If you want a full picture of site structure, you have to manually check the global settings page and cross-reference taxonomy tags one by one.

With this connection, your AI agent handles all those steps for you. You just ask it to list published posts or pull author details. The data appears clean and structured right inside your chat window. It removes the tedious multi-tab manual process completely.

The Ghost MCP gives you full visibility into content structure.

You never have to manually check if a new post uses an existing tag or if the site's primary title schema has changed. You can call list_content_tags and get_site_settings in rapid succession, building your data model on demand without leaving your current workspace.

Your agent now sees Ghost as one cohesive data layer. It’s not a series of tabs; it's a single source of truth you can query.

Ghost CMS MCP: 11 Tools for Publishing Data

Use these tools to read structured data from your Ghost CMS, whether you need to list posts, check author details, or pull global site settings.

#	TOOL	DESCRIPTION
01	`verify_api_connection`	Checks if the connection between your AI agent and Ghost is working correctly.
02	`get_post_by_slug`	Fetches all details for a specific blog post using its unique URL slug.
03	`get_author_details`	Retrieves the full profile information and bio for a specified author.
04	`get_page_by_slug`	Gets details about non-blog pages (like 'About Us') by their slug.
05	`get_site_settings`	Accesses global site configurations, such as routing rules and title schemas.
06	`get_tag_details`	Retrieves metadata for a specific content category or tag.
07	`list_subscription_tiers`	Provides a list of all active membership levels and their associated pricing details.
08	`list_blog_authors`	Lists every author who has contributed content to the publication.
09	`list_static_pages`	Retrieves a list of all pages that are not blog posts, including their slugs and titles.
10	`list_published_posts`	Lists metadata for every article currently live on the site.
11	`list_content_tags`	Retrieves a comprehensive list of all available content categories and tags used across the site.

See It in Action

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

U List the 5 most recent published posts.

Retrieving posts... I found 5 recent articles, including 'Welcome to Ghost' and 'Writing with Markdown'. Would you like to read the full content of any of them?

U Get the content of the post with the slug 'welcome'.

Fetching post... The post 'Welcome' was published on October 1st. The main content is: 'Welcome, it's great to have you here...'.

U List all active subscription tiers.

Fetching tiers... Your publication has 2 active tiers: 'Free' ($0/month) and 'Premium' ($5/month).

Frequently Asked Questions

01 **How do I use the Ghost MCP to list all published posts?**

You ask your agent to list_published_posts. It will return a list of metadata for every article, including titles and publication dates.

02 **Can I find out which tags are available using Ghost MCP?**

Yes. Use the list_content_tags tool. This gives you a complete inventory of all categorization tags applied across your site.

03 **How does get_author_details work with the Ghost MCP?**

You provide the author's name or unique identifier, and the tool retrieves their full profile details, including bio and contribution history.

04 **Do I need to manually enter my API key after connecting the Ghost MCP?**

No. The connection process requires you to provide your Base URL and Content API Key once during setup. After that, your agent handles authentication automatically.

05 **What is the difference between list_static_pages and list_published_posts?**

list_published_posts only lists articles (blog posts). list_static_pages lists non-article pages like 'Contact' or 'About Us'.

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 `"ghost": { "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

Ghost 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 Ghost. 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	Ghost MCP
Server ID	`019d75a4-4cfb-7318-bb5e-a6e7e9d6ea0c`
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/ghost.