MCP SERVER | NO CODE | CLOUD HOSTED

Logseq Knowledge MCP

Talk to your notes like a database.

Logseq Knowledge Management MCP connects your personal outliner graph to any AI client. It lets you treat your notes like a database, using natural conversation to read, write, and organize structured content across pages and blocks while keeping everything local and private.

A+ Quality Score 98.33/100

bi-directional-linking | markdown | knowledge-graph | privacy-first | outliner

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/logseq-knowledge-management

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.

Logseq (Knowledge Management) MCP

10 tools available
Cloud-hosted on Vinkius

This connector gives your agent direct access to the internal structure of your Logseq instance. You can manage your personal knowledge graph by simply talking to it. Forget copy-pasting huge chunks of notes into a prompt just to get an overview; your AI client now sees the entire hierarchy, down to the individual block level.

Need to track research threads or technical documentation? Your agent doesn't just read text; it understands that 'Project Alpha' is a page and that 'Verify API endpoints' is a specific sub-task (a block) within it. You can ask it to list all pages, find deeply nested project structures, or even delete old metadata loops securely.

Whether you're an engineer managing tech specs or a researcher tracking academic ideas, this MCP lets your agent perform complex operations like updating properties on specific blocks or running deep text searches across the entire vault. Vinkius makes connecting to this powerful local system easy; just connect once from any compatible client and get full control of your graph structure through conversation.

Core Capabilities

01 — List all pages

It gives you an immediate list of every page in your Logseq vault.

02 — Retrieve specific page details

You can pull the metadata for any single page using its name or UUID, giving you targeted information.

03 — Create and manage pages

The agent can deploy new organized pages into your graph or delete them completely.

04 — Inspect hierarchical blocks

It extracts the complete, nested structure of an outliner tree from a specific page.

05 — Modify individual content blocks

The agent can append new thoughts to existing blocks, update their properties, or remove them safely.

06 — Search the entire knowledge base

You run local queries that find specific text targets across every page and block in your graph.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/logseq-knowledge-management** — connect your AI agent in three steps.

01 Subscribe to this MCP on Vinkius, then enable the HTTP API within your Logseq settings.

02 Enter your unique Logseq API Token and the Host URL into your agent client's configuration.

03 Start asking questions or giving commands. Your AI client communicates directly with the local graph structure for results.

The bottom line is, you get to manage your complex knowledge base using simple natural language prompts instead of navigating menus and copy-pasting data.

Built For

This MCP targets the power user who treats their notes like a structured database. It's for researchers, developers, and writers whose work relies on connecting disparate pieces of information scattered across dozens of pages.

Knowledge Worker

You use it to synthesize meeting transcripts or research papers by asking your agent to list all related concepts and retrieve the specific blocks where those ideas were mentioned.

Software Developer

You manage technical documentation, running queries through the MCP to find old API usage examples or update project log blocks without opening a dedicated dev tool.

PKM Enthusiast

You audit your graph structure by asking it to identify deeply nested relationships between unrelated ideas, ensuring your knowledge base stays clean and optimized.

What Changes When You Connect

01 You never have to manually copy-paste context again. Instead, you simply ask your agent to search the entire graph using `search_content`, and it delivers precisely what you need, instantly.

02 Maintain privacy while working with complex data. Because this MCP connects locally, all your notes stay within your private Logseq environment, giving you full control over your knowledge structure.

03 Keep your project files organized by automating page creation or deletion. You can use `create_page` to start a new topic or `delete_page` when the work is done.

04 Manage granular details using block tools. If you need to refine a specific point, you don't have to rewrite the whole thing; just ask your agent to `update_block` on that single item.

05 Understand complex relationships by viewing outliner trees. The `get_page_blocks` tool lets you see exactly how deeply nested your ideas are structured without manual inspection.

Real-World Applications

Synthesizing meeting notes across multiple days

A project manager needs to find all instances of 'Q3 budget' mentioned in the last month's worth of notes. Instead of opening 15 different pages, they ask their agent to `search_content` for that phrase and get a list of every relevant block across the entire vault.

Refining a specific technical concept

A developer wants to add a new API endpoint reference to an existing project plan page. They ask their agent to `insert_block` with the details, ensuring the new info is perfectly nested under the right section without messing up the formatting.

Archiving old research projects

A student has completed a thesis and needs to clean up related pages. They ask their agent to `delete_page` for the entire 'Draft 2023' section, permanently removing all associated metadata loops.

Mapping complex ideas

A writer is trying to understand how three separate concepts (e.g., quantum physics, medieval history, and supply chain logistics) are connected in their vault. They use `get_page_blocks` on a key page to map the exact structural relationships between these different topics.

Patterns to Avoid

Treating notes like unstructured text

✗ AVOID

Pasting a huge wall of text into your agent and asking, 'What are the main points?' The agent gives you generic summaries that miss the internal connections.

✓ INSTEAD

Instead, ask your agent to use `get_page_blocks` on the specific page. This shows the structured hierarchy, allowing the AI to understand which ideas are related sub-tasks and which are standalone notes.

Losing track of updates

✗ AVOID

Manually updating a block in Logseq, but forgetting to remember that you changed something. You can't prove when or how the change happened.

✓ INSTEAD

Use `update_block`. This tool safely modifies raw properties and content for a specific block while retaining all necessary linking indices, keeping your notes accurate.

Overwriting critical sections

✗ AVOID

Trying to paste an entire section of new text over existing data because it's faster. You risk losing the original structure and links.

✓ INSTEAD

Use `insert_block` instead. This appends the content cleanly, ensuring that the new material is added as a distinct, traceable unit within your graph.

The Right Fit

Use this MCP if your primary need is structural integrity and deep querying of *organized* knowledge. If you are working with outliners, linked concepts, or hierarchical notes, this is for you because it gives the agent access to blocks and relationships, not just text blobs. Don't use it if you simply want a general-purpose chatbot that answers questions based on random web pages; those tools aren't aware of your local graph structure. If your workflow involves writing simple, linear documents with no internal links or nested tasks, you probably don't need this level of granularity.

The Pain of Manual Knowledge Retrieval

Today, finding a specific piece of information means manually opening the 'Project X' page, scrolling through meeting notes, and hoping that the relevant block hasn't been buried under three levels of unrelated thoughts. You're spending time hunting for UUIDs or cross-referencing dates just to prove a point.

With this MCP, you tell your agent what you need —like 'all blocks related to Q3 marketing budget'— and it handles the deep graph query. It doesn't matter if that data is spread across pages or buried in deeply nested outliner structures; you get the precise results immediately.

Structured Knowledge Operations with Logseq MCP

Previously, modifying your knowledge required navigating to the correct page and manually executing changes. If you needed to update a project scope block or delete an entire draft section, it was a multi-step process prone to human error.

Now, all those steps are abstracted away. You simply prompt your agent with the command —'update the scope block for Project Alpha'—and the MCP executes the precise change, preserving every link and index in the background.

Logseq (Knowledge Management) 10 Tools

Use these tools to programmatically interact with your Logseq vault. You can list pages, update blocks, run deep searches, and manage the structure of your personal knowledge graph.

#	TOOL	DESCRIPTION
01	`list_pages`	Retrieves a list of all existing pages in your Logseq vault.
02	`get_page`	Pulls the detailed metadata for one specific page using its name or unique ID.
03	`create_page`	Creates a brand new, organized page within your local Logseq graph.
04	`delete_page`	Irreversibly removes an entire active Logseq page and all its content.
05	`get_page_blocks`	Extracts the complete, nested outliner tree structure from a selected page.
06	`insert_block`	Appends a new, managed block of text to any specific location within your graph.
07	`update_block`	Modifies the raw properties and content of an existing tracked block while preserving its links.
08	`delete_block`	Permanently removes a specific outliner block, including any child dependencies.
09	`search_content`	Runs local queries to pull out all text matches across your entire knowledge base.
10	`get_current_graph`	Validates the current environment by identifying active graph paths and database directories.

See It in Action

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

U Search my Logseq graph for 'smart building research'

I've scanned your graph and found 3 relevant matches: 1) Page 'Smart-Buildings-Project', 2) Block in 'Journal 2026-03-27' mentioning sensor setups, 3) Tag '#research' in multiple blocks. Would you like to see the blocks from the project page?

U Create a new page called 'Meeting Notes' with content '# Meetings 2026'

Page 'Meeting Notes' created successfully in your graph. I've added the initial markdown header. You can now use the `insert_block` tool to add specific meeting items or agenda points.

U Add a block to the 'Project Alpha' page: 'Verify API endpoints for production'

Block added successfully to 'Project Alpha'. It is now part of the outliner tree for that page. Would you like me to add any specific child blocks or sub-tasks under this point?

Frequently Asked Questions

01 **How does Logseq Knowledge MCP maintain my data's privacy?**

The connection is local. Your agent client talks directly to your private Logseq instance via an API token, meaning your notes never leave your controlled environment.

02 **Can the Logseq Knowledge MCP handle large vaults?**

Yes, because it queries the internal graph structure rather than relying on external indexing. It's designed to manage complex, large-scale outliner data efficiently.

03 **What is the difference between `search_content` and `get_page`?**

`get_page` retrieves all metadata for one specific page by name or ID. `search_content` runs a query across your whole vault to find mentions of certain text targets, regardless of which page they live on.

04 **Does the Logseq Knowledge MCP let me add images?**

The focus is on structured data and outliner blocks. While it manages markdown content well, its primary function isn't handling multimedia files.

05 **If I delete a block using `delete_block`, does it affect other things?**

No. The MCP is designed to safely remove explicit nodes while retaining the integrity of surrounding links and indices, ensuring nothing else breaks because of the deletion.

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 `"logseq-knowledge-management": { "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

Logseq (Knowledge Management) 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 Logseq (Knowledge Management). 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	Logseq (Knowledge Management) MCP
Server ID	`019d75c9-d2d1-701a-8821-f0bc2d3b91b9`
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/logseq-knowledge-management.