MCP SERVER | NO CODE | CLOUD HOSTED

Omnivore (Read-Later) MCP

Search, summarize, and act on everything you've ever saved.

Omnivore (Read-Later) connects your saved articles and links to any AI client. Use this MCP to manage your personal reading list directly through conversation, letting your agent search, summarize, and categorize knowledge from all your curated sources.

F Quality Score 3.6/100

read-it-later | content-curation | bookmarking | article-retrieval | knowledge-base | search-indexing

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/omnivore-read-later

SCAN TO VIEW

The connectivity layer between AI and the world's software.

Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the connectivity layer where AI connects 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.

Omnivore (Read-Later) MCP

4 tools available
Cloud-hosted on Vinkius

You keep hundreds of interesting web links in read-it-later apps, but finding that one specific piece of information later is a pain. This connector turns your entire Omnivore library into an active source for your AI agent. Your agent can find content based on labels or folders; it doesn't just show you a list of titles, it retrieves the full text and author details so you can analyze it immediately. You can also instantly save new articles from any website without leaving your chat window. If you use Vinkius to connect this MCP, you gain an immediate knowledge base that lets your AI agent act on everything you've ever read.

Core Capabilities

01 —

02 —

03 —

04 —

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/omnivore-read-later** — connect your AI agent in three steps.

01 Subscribe to this MCP on Vinkius and enter your Omnivore API Key.

02 Connect your preferred AI client (Claude, Cursor, etc.) to the active catalog link.

03 Direct your agent: ask it to search for a topic or save a new URL from your conversation.

The bottom line is that you treat your entire reading list like an indexed database you talk to naturally.

Built For

Researchers, journalists, and content creators who accumulate massive amounts of source material need this. If you ever find yourself manually copying links into a spreadsheet just to remember what they were about, this MCP fixes that.

Researcher

Uses the connector to search for specific concepts across dozens of saved academic papers or articles within their private library.

Writer/Journalist

Quickly saves source material and then uses the MCP to retrieve full content for inspiration or citation checks during a drafting session.

Knowledge Worker

Manages a constantly growing library of industry news, ensuring nothing valuable gets lost in bookmarks folders.

What Changes When You Connect

01 Instead of opening multiple tabs to search your bookmarks, the `search_articles` tool lets your agent filter your whole library using labels or status. You just ask for what you need.

02 Need deep context? Use `get_article` to pull the complete text and metadata from a saved piece, allowing your AI client to summarize key points directly in the chat.

03 When you stumble upon an interesting article while working, use `save_url`. You instantly add it to your library without ever having to switch applications. It's built right into your conversation flow.

04 You can confirm everything is set up correctly using `get_me` to verify your account details and make sure the connection works before you start processing data.

05 It turns a passive collection of links into an active knowledge base, letting you retrieve source material on demand when writing or researching.

Real-World Applications

Reviewing old research notes

A researcher needs to write a literature review on AI ethics. Instead of manually searching files, they ask their agent to use `search_articles` for 'AI' AND label:ethics AND is:unread. The agent finds the top five most relevant sources and pulls the full content using `get_article` so they can start drafting immediately.

Saving inspiration during a meeting

A marketer hears about a new trend and wants to save an article link. They use their agent to call `save_url`, passing the URL in real time, ensuring the source material is captured for later brainstorming without taking notes.

Checking content gaps

A content creator has a project scope and wants to verify if they've saved enough examples. They ask their agent to search using `search_articles` filtered by 'topic:UX design' and check the count, ensuring they have sufficient source material.

Getting quick context on an old article

A user remembers reading a great piece from two months ago but can't recall where. They ask their agent to search for keywords, and the agent uses `search_articles` followed by `get_article` to pull up the full text, solving the mystery instantly.

Patterns to Avoid

Treating it like a bookmark manager

✗ AVOID

Simply copying and pasting links into a document and hoping for the best. You lose context, labels, and searchability.

✓ INSTEAD

Instead, let your agent use `save_url` to capture the link, then use `search_articles` to filter it by tags or reading status within your Omnivore library.

Forgetting what you read

✗ AVOID

Spending hours going through folders labeled 'To Read' and having no idea which article was most relevant at the time.

✓ INSTEAD

Use `search_articles` to filter for articles by specific labels, like 'actionable' or 'must-read', so your agent surfaces only the high-signal content.

Manually extracting article details

✗ AVOID

Having to visit every single link in Omnivore just to copy the author and title into a spreadsheet.

✓ INSTEAD

Use `get_article` through your agent. It pulls the full content, author, and labels automatically for immediate analysis.

The Right Fit

Use this MCP if you have large quantities of saved web links (articles, blog posts) and need to treat them like an active knowledge database that can be searched and summarized on demand. If your goal is simply organization—like moving articles between folders—then a dedicated bookmarking app works fine. However, if the goal is *analysis*—you want your AI agent to read the full text retrieved by `get_article`, summarize it, or compare it against other sources found via `search_articles`—this MCP is essential. Don't use this if you only need a simple list of links; use it when you need actionable knowledge extraction.

The endless cycle of saving and forgetting content

We all do it: we find an incredible article, save the link, feel smart for curating our reading list. But then life gets busy. Weeks later, when we need that specific insight—say, about supply chain logistics—we open our bookmark manager and are faced with hundreds of titles. We have to manually scroll through folders labeled 'Business' or 'Science,' opening links one by one just to remember the key takeaway.

With this connector, your agent handles the friction. You ask it a question about your saved knowledge base. It uses tools like `search_articles` to pinpoint exactly what you need and then pulls the full text using `get_article`. The result isn't a list of links; it’s an answer.

Getting actionable insights with Omnivore (Read-Later) MCP

The biggest manual step that goes away is the cross-referencing. You don't have to open ten different articles and copy three key quotes from each one into a single document for comparison. Your agent does it all in one conversation.

Now, your reading list isn't just storage; it’s an integrated resource. It allows you to perform deep analysis on every piece of content without leaving the chat window.

Omnivore (Read-Later): 4 Tools

These four tools allow your agent to search your library, retrieve full article content, save new links, and manage your user profile directly within the chat interface.

#	TOOL	DESCRIPTION
01	`get_article`	Fetches the full text and details for one specific article you saved.
02	`get_me`	Checks your current Omnivore account details and confirms your connection status.
03	`save_url`	Adds a new web link to your library instantly, without opening the Omnivore website.
04	`search_articles`	Filters and finds articles across your entire collection using labels or reading status filters.

See It in Action

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

U Search my Omnivore library for unread articles about 'Machine Learning'.

I found 3 unread articles about Machine Learning. The most recent is 'Understanding Transformers' (slug: understanding-transformers). Would you like me to fetch the full content of any of these?

U Fetch the full content of the article with slug 'mcp-guide' for username 'alex_dev'.

I've retrieved the article 'The Ultimate MCP Guide'. It covers the Model Context Protocol architecture and implementation details. Would you like a summary of the key points?

U Save the URL 'https://blog.omnivore.app/p/getting-started' to my library.

Successfully saved the URL to your Omnivore library. It is now available in your inbox for later reading.

Frequently Asked Questions

01 **How do I use Omnivore (Read-Later) MCP to search my articles?**

You tell your agent to find what you need. It uses `search_articles` and accepts filters like labels or read status, letting you narrow down thousands of links to just the relevant ones.

02 **Can I save new links using Omnivore (Read-Later) MCP?**

Yes. You can use `save_url` right in your chat conversation to instantly add a link to your library without needing to visit the Omnivore site.

03 **What does get_article do with Omnivore (Read-Later) MCP?**

The `get_article` tool retrieves the complete, full text of an article. This is crucial because it lets your agent summarize or analyze content that you can't just skim from a title.

04 **Do I need to use Omnivore (Read-Later) MCP with all my clients?**

No, you only connect the MCP to the AI client you prefer. Vinkius lets your agent work whether it's connected from Cursor or Claude.

05 **Is there a tool to check if my connection is working with Omnivore (Read-Later) MCP?**

Yes, use `get_me`. This simple tool checks your account details and confirms that the API key you provided is active and connected.

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 `"omnivore-read-later": { "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

Omnivore (Read-Later) 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 Omnivore (Read-Later). 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	Omnivore (Read-Later) MCP
Server ID	`019e38cb-c9ce-707f-82e6-8778d1af053a`
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/omnivore-read-later.