MCP SERVER | NO CODE | CLOUD HOSTED

Feedly MCP

Control your entire news aggregation workflow.

Feedly connects your entire news aggregation workflow to any AI agent. It lets you manage every detail of your curated content, from listing all personal collections and tags to pulling the latest articles from specific streams or boards directly through natural conversation.

A+ Quality Score 100/100

rss-aggregation | news-monitoring | content-curation | knowledge-management | stream-processing

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

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.

Feedly MCP

12 tools available
Cloud-hosted on Vinkius

Managing a high volume of information across multiple sources used to mean switching tabs constantly and manually bookmarking everything. Now, connect Feedly via this MCP and take full control of your news flow with simple chat commands. You can list all your curated collections and feeds, then pull the latest articles from specific streams or entire categories. Need to save an article for later? Mark it as read or stash it on a board without touching the web UI. If you're using Vinkius, you get access to this tool alongside thousands of others, letting your agent handle everything from content discovery—searching across the whole Feedly index for new topics—to accessing your profile metadata. It’s all about keeping your focus on work, not on where the source material is stored.

Core Capabilities

01 — Discovering new sources

Search the entire Feedly index to find relevant RSS feeds or trending topics you didn't know existed.

02 — Curating content lists

List all existing collections, boards, and tags to see how your information is currently organized.

03 — Retrieving article batches

Fetch the latest articles from an entire feed stream, a specific board, or content filtered by a tag.

04 — Managing read status

Mark articles as read directly through your agent, keeping track of what you've already processed.

05 — Accessing user data

Retrieve your current Feedly profile and subscription metadata for context during research sessions.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and enter your specific Feedly Developer Access Token.

02 Connect the credential to your AI client, like Cursor or Claude.

03 Ask your agent directly what you need—for example, 'Show me all my tech collections' or 'Find articles tagged with quantum computing'.

The bottom line is that your AI agent handles all the API calls and data formatting so you just get a clean list of information in your chat window.

Built For

This MCP is essential for content professionals, researchers, and knowledge workers who deal with dozens of disparate news sources daily. If you spend more time organizing articles than actually reading them, this tool saves hours.

Research Analyst

Needs to quickly pull the latest industry updates or trending topics from multiple sources into a single research document without leaving their chat interface.

Content Strategist

Monitors dozens of competitor feeds and niche blogs, saving inspiration or raw data directly onto specific Feedly boards for later use in content creation.

Journalist/Writer

Needs to check multiple source streams across different categories and mark articles as read immediately after drafting a story segment.

What Changes When You Connect

01 You can quickly pull the latest industry news and trending topics by using `get_stream_contents` to check specific feeds or categories.

02 Organize massive amounts of data easily. Use `list_collections` and `list_boards` to see your entire structure before pulling articles with `get_board_contents`.

03 Never lose track of a good source again. You can use the agent to find new sources using `search_feeds`, or narrow down content by running `get_tag_contents`.

04 Keep your reading list clean and accurate. The `mark_as_read` tool lets you update article status without opening any browser tabs.

05 Streamline research context. You can check your full profile details using `get_profile`, which provides valuable metadata for your reports.

Real-World Applications

The Quarterly Industry Review

A researcher needs to pull the top 10 articles on 'sustainable energy' from three different sources. Instead of logging into Feedly and clicking through, they ask their agent to use `get_stream_contents` for each source, giving them a clean, consolidated list ready for summarization.

Capturing Project Inspiration

A content strategist finds several design examples from niche blogs. They tell their agent to search using `search_feeds` to find those sources, then use the board functions like `get_board_contents` to collect all relevant articles onto a 'Q3 Design' board.

Tracking Competitor Moves

A marketing manager wants to know which competitors are posting about 'AI ethics'. They ask their agent to use `get_tag_contents` with the tag 'AI ethics,' immediately pulling all relevant content from multiple streams.

Pre-Meeting Prep Work

Before a meeting, you need context on your own subscription status. You ask your agent to run `get_subscriptions` and then follow up with `list_collections` to confirm exactly what sources were reviewed.

Patterns to Avoid

Assuming a source exists

✗ AVOID

The user tries to ask for 'The new space exploration feed' but doesn't know the exact name or category, leading to an error.

✓ INSTEAD

First, run `list_collections` to see all your established categories. If that fails, use `search_feeds` and tell your agent what keywords you are looking for.

Trying to manually update status

✗ AVOID

The user reads 50 articles on a desktop feed and then has to go through the UI just to mark them as read, wasting time.

✓ INSTEAD

Just tell your agent, 'Mark these 50 articles in this stream as read.' It handles the update via `mark_as_read` instantly.

Searching too broadly

✗ AVOID

The user asks for 'all content' without specifying a time frame or source, resulting in an overwhelming and unusable data dump.

✓ INSTEAD

Be precise. Instead of general queries, specify the scope: use `get_board_contents` for articles saved to a specific board ID.

The Right Fit

Use this MCP if your core problem is content aggregation—you have dozens of sources and you need an organized way to read, tag, save, and track those individual pieces. This tool excels at retrieving structured data based on source location (boards, tags, streams). Don't use it if your primary goal is messaging or collaboration; for that, you need a chat-based MCP. Similarly, don't use this if you just need general knowledge—if the information lives in unstructured documents, you might need an RAG-type tool instead of one focused on RSS feeds.

When to use it: When your workflow is 'Consume -> Organize -> Analyze.' Use `list_tags` or `get_tag_contents` to gather related ideas across multiple sources. When to avoid it: If you just need to send a quick reminder, use a messaging MCP. This tool lives in the content management space; its job is to bring the articles to you, not to talk to your team about them.

The manual process of keeping up with industry news feels like a full-time job.

Today, staying current means opening 8 different tabs. You skim the headlines on Source A, copy a link to your Notion board. Then you jump to Source B and manually write down key takeaways. By the time you've processed ten sources, you’re exhausted, and half the valuable links are buried in bookmarks that never get opened.

With this MCP, your agent handles the entire process. You just tell it what information you need —say, 'Give me all articles on quantum computing from my top three feeds.' It pulls the data directly, giving you a clean, summarized feed without any manual copying or tab-switching.

Feedly MCP: Organizing Your Content Streams

You no longer have to manually check every category to see if something new has dropped. Instead, you ask the agent to run

The difference is that your AI client acts as a central dashboard for all your feeds. You get actionable data immediately, without leaving your

`list_collections` and then specify which stream needs attention. The agent pulls everything for you in one step.

primary workspace or manually clicking through dozens of source websites.

Feedly MCP: 12 Tools for Content Management

These tools give you granular control over every part of your Feedly account, from listing all boards to pulling specific article details.

#	TOOL	DESCRIPTION
01	`get_board_contents`	Pulls all the stored articles from one of your specific Feedly boards.
02	`get_entry`	Retrieves full details for a single, specific article entry by its ID.
03	`get_profile`	Fetches your current Feedly user profile information and settings.
04	`get_stream_contents`	Retrieves the latest articles from a designated feed, category, or global stream.
05	`get_subscriptions`	Lists all the individual feeds you are currently subscribed to in Feedly.
06	`get_tag_contents`	Pulls articles that have been assigned a specific tag, allowing focused retrieval.
07	`list_boards`	Generates an inventory of all the personal boards you've set up for saving articles later.
08	`list_collections`	Lists every category and curated collection within your Feedly account.
09	`list_tags`	Provides a list of all the unique tags you have used across your content.
10	`mark_as_read`	Updates the status of specific articles, marking them as read in Feedly.
11	`search_feeds`	Searches the entire Feedly index to locate brand new RSS feeds relevant to your interests.
12	`search_topics`	Searches for general, trending subjects or specific keywords across the platform's content base.

See It in Action

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

U List my Feedly collections.

I've retrieved your collections. You have 'Tech News', 'Design Inspiration', and 'Marketing' categories. Which collection's latest articles would you like to see?

U Show me the latest 5 articles from the 'Tech News' category.

Fetching stream contents... Here are the latest updates including 'AI Breakthroughs in 2024' and 'New Smartphone Launch'. Would you like me to summarize any of these for you?

U Search for feeds about 'Edge Computing'.

Searching the Feedly index... I found several relevant feeds including 'The Edge Blog' and 'Computing Weekly'. Would you like the stream IDs to add them to your collections?

Frequently Asked Questions

01 **How does the Feedly MCP help me discover new topics?**

You can use `search_feeds` to search the entire Feedly index for new RSS feeds. This tool helps you find relevant sources without knowing their exact URL or category.

02 **Can I organize articles using the Feedly MCP?**

Yes, you can list all your tags with `list_tags` and then use `get_tag_contents` to pull every article associated with that tag across multiple feeds.

03 **What is the difference between `list_collections` and `list_boards`?**

Collections are your primary categories or overarching source groups. Boards are specific, curated lists of articles you save for later viewing or reference.

04 **How do I update my status using the Feedly MCP?**

You simply ask your agent to mark articles as read. It executes the `mark_as_read` tool, updating the article status across your account instantly.

05 **Do I need an API key for the Feedly MCP?**

Yes. You must subscribe and enter a valid Feedly Developer Access Token so your agent can authenticate and perform actions on your behalf.

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 `"feedly": { "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

Feedly 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 Feedly. 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	Feedly MCP
Server ID	`019d7597-f5b9-73c0-b342-55d54bab4fc1`
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/feedly.