MCP SERVER | NO CODE | CLOUD HOSTED

Amplience MCP for AI Agents

Manage Digital Asset Content and Schema Deployments

Amplience connects your content management system directly to your AI agent. This MCP lets you query complex content structures, pull metadata, update schemas, and push finished articles or product listings live—all through conversation.

A+ Quality Score 100/100

digital-asset-management | content-architecture | edge-delivery | schema-management | enterprise-content

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

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.

Amplience MCP

10 tools available
Cloud-hosted on Vinkius

Stop clicking through tabs just to manage content. Amplience links your headless CMS to any intelligent AI client, letting you treat enterprise content architecture like a chat session. You can tell the agent exactly what needs changing: 'Update the hero banner's call-to-action and push it live.' The system handles fetching the current structure, verifying schemas, making the necessary modifications, and deploying the changes directly to your edge delivery API. Because this MCP is hosted on Vinkius, you connect once from any compatible client—like Claude or Cursor—and get immediate access to managing content across your entire digital ecosystem. It's about using natural language to perform complex publishing tasks.

Core Capabilities

01 — List all Amplience Hubs and Repositories

See every accessible workspace and the main content repositories available to your agent.

02 — Get Content Item Details (Schema & Payload)

Retrieve a full copy of any single piece of content, including its current schema definition and metadata for review.

03 — Create or Modify Structured Content

Generate entirely new drafts or change existing data items using structured JSON payloads that match the required schemas.

04 — Manage Deletion of Assets

Permanently remove content items from the repository database after validating their version status.

05 — Publish Content to Live CDN

Execute the final step by pushing approved and modified content versions directly to your live delivery network.

One Click on Vinkius — From Prompt to Execution

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

01 Enable the Amplience connector within your client environment.

02 Provide your specific Amplience Hub Name along with a Personal Access Token.

03 Use natural language prompts in your AI client to guide the agent through content retrieval, modification, and final publishing.

The bottom line is: You talk to your AI client using plain English commands, and it executes complex CMS operations against Amplience for you.

Built For

This MCP is for technical teams who spend too much time in the CMS UI. If your job involves translating marketing ideas into structured content, or pushing changes from a database to a live website, this tool saves hours of manual clicking.

Content Marketing Manager

Uses the agent to draft promotional blog posts and populate fields using specific SEO guidelines before assigning them for review.

Frontend Developer

Asks the agent to read an Amplience schema's output validation rules directly in the IDE, allowing them to scaffold matching React components instantly.

E-commerce Operations Specialist

Commands the agent to locate and update specific promotional banners or product metadata across different repositories for a seasonal sale.

What Changes When You Connect

01 Eliminate manual content audits. Use `list_content_items` to pull paginated lists of articles or product blocks, instantly giving you a full inventory of your repository.

02 Accelerate publishing cycles. Instead of copying data into the CMS UI, use `create_content_item` to generate structured drafts based on defined schemas with simple chat prompts.

03 Maintain technical accuracy. When reviewing content, `get_content_item` pulls the precise schema lock and payload, guaranteeing your AI agent works with the current version rules.

04 Control the live deployment. After drafting changes, use `publish_content_item` to push approved content directly to the CDN edge without manual website publishing steps.

05 Build reliable developer workflows. Frontend teams can read a specific schema's output validation structure by running `get_content_item`, which allows them to scaffold accurate code interfaces.

Real-World Applications

Updating Global Sale Banners

An e-commerce specialist needs to change the Black Friday banner across three different product lines. They ask their agent to locate all relevant items, use `list_content_items` to confirm they found every instance, and then instruct the agent to execute an `update_content_item` for the new dates.

Building a New Blog Series

A marketing manager needs 10 drafts for a technical blog series. They prompt their agent to use `create_content_item` ten times, feeding it structured JSON payloads that adhere to the official 'Blog Post' schema.

Cleaning Up Old Content

The team realizes an old product line is deprecated. They ask the agent to search for all content items related to that line using `list_content_items`, and then execute `delete_content_item` on the confirmed list.

Verifying Live Site Status

A developer suspects a recent product update isn't showing up correctly. They ask the agent to use `get_delivery_content` to pull the exact structural matching for the current live CDN blocks and verify the output.

Patterns to Avoid

Publishing without checking schemas

✗ AVOID

A developer attempts to submit a new content item payload directly, but it fails because the payload doesn't match the required field types or structure for that specific repository.

✓ INSTEAD

First, use `get_content_item` to pull the exact schema and validation rules. Then, ensure your generated JSON perfectly matches those requirements before running `create_content_item`.

Manually listing all content

✗ AVOID

A marketer has 50 articles in a repository and must manually navigate through the CMS interface to confirm which ones need review.

✓ INSTEAD

Use `list_content_items` with your agent. It pulls paginated data for you, giving you a clean list of all content items instantly.

Confusing draft status and live publishing

✗ AVOID

A team member edits the content in a private sandbox but forgets to run the final step, leaving the changes only visible internally.

✓ INSTEAD

After any modifications or creations, always instruct your agent to use `publish_content_item` to push the changes from draft status to the live CDN.

The Right Fit

Use this MCP if your content process requires full lifecycle control: creating, modifying, and publishing structured assets. If you need to see what repositories exist or list all folders, use `list_hubs` and `list_repositories`. However, don't use this MCP just because you want to read a simple piece of data; for that, a general API query is faster. Also, if your goal is only auditing content without modifying it, stick to `list_content_items` and avoid the write tools like `update_content_item`. This tool gives deep control over *how* content moves from draft to live.

Amplience MCP: Managing Content Architecture with AI Agents

Today, updating a single product description or changing a promotional banner requires logging into the CMS, navigating deep folder structures, manually finding the right content item, making sure you're on the correct revision, and then hitting multiple save buttons. It’s click-heavy, error-prone, and takes too long.

With this MCP, your agent handles all that complexity in a conversation. You simply tell it what to change—say, 'Update the hero banner CTA.' The agent finds the right item, validates its schema, makes the modification, and hands you the result.

Amplience MCP: Deploying Structured Content Updates with AI Agents

The biggest time sink is ensuring that every piece of content—whether it's a blog post or an e-commerce listing—has the correct, up-to-date metadata before it goes live. You spend hours cross-referencing drafts against the latest schema requirements.

Now, you can command your agent to generate and validate payloads using `create_content_item` or `update_content_item`, making sure that every single field is correctly typed and structured before you ever push it out.

Amplience MCP: 10 Tools for Content Repository Operations

Use these tools to manage the entire lifecycle of your digital assets—from listing repositories to pushing finished content live.

#	TOOL	DESCRIPTION
01	`create_content_item`	Generates an entirely new, structured content piece that matches a defined schema within a specific folder.
02	`delete_content_item`	Permanently removes a content item from the repository database after validating its version status.
03	`get_content_item`	Retrieves all configuration data for a single content item, including its schema revision lock.
04	`list_content_items`	Pulls paginated lists of content items from an entire repository for bulk review or auditing.
05	`get_delivery_content`	Retrieves the exact structural data needed to verify what is currently being served by the Delivery CDN blocks.
06	`list_folders`	Lists all organizational folders within a specific content repository.
07	`list_hubs`	Shows every active workspace (Hub) you have access to in Amplience.
08	`list_repositories`	Lists all content repositories that exist within a chosen Hub.
09	`publish_content_item`	Pushes the latest, modified version of a specific content item directly to the live delivery CDN.
10	`update_content_item`	Modifies an existing content item's data structure while ensuring it matches its current schema requirements.

See It in Action

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

U What are all the active repositories in my main Hub? I need to know where the product data lives.

Hub: Main Ecommerce Portal

- 🛒 Product Catalog (ID: `6k33i1`)
- ✍️ Blog Content (ID: `6m75o9`)
- 🖼️ Marketing Banners (ID: `9q82y3`)

You can now ask me to list the folders within any of these for more detail.

U Pull the structural metadata and schema lock for the 'Spring 2024' banner item.

Content Item: Spring Banner (ID: `5tYv92`)

- Schema Lock Version: 5
- Payload:

```json
{
"title": "Spring Sales!",
"href": "/shop/spring"
}
```

This means you can change the title or link, as long as they match this structure.

U I just edited the product data. Can you publish it to the live CDN now?

Confirmed. Publishing content item '5tYv92' now.

✅ **Status:** Published successfully.

- **Target:** Live Delivery CDN
- **Timestamp:** 2026-06-12T14:30:00Z
- **Verification:** The content is live and visible at the root domain.

Frequently Asked Questions

01 **How can I use Amplience MCP to update my website content without logging into Amplience?**

You don't have to log in. Your AI agent handles all the connection details and API calls for you. You just tell it what needs changing, and the changes get pushed live automatically.

02 **Does Amplience MCP help me find out what content schemas I need to follow?**

Yes. The agent can pull the structural metadata (schema lock) for any item you name. This gives you the exact JSON structure and field rules required before you try to create or update anything.

03 **Can Amplience MCP manage content across multiple websites or hubs?**

The MCP is designed to read all accessible Hubs, so it can help you audit and manage content assets across various environments within your enterprise setup.

04 **If I edit a banner using Amplience MCP, will that change show up on my live site?**

Not until you tell the agent to publish it. You must use the publishing tool to push the changes from draft status out to the active delivery CDN.

05 **Can I generate brand new content drafts using Amplience MCP?**

Yes, you can instruct your AI client to create completely structured content items. You provide the data, and the agent ensures it adheres to the correct schema before generating a draft.

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 `"amplience": { "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

Amplience 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 Amplience. 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	Amplience MCP
Server ID	`019d754d-eb67-7197-8165-6632cbc1033f`
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/amplience.