MCP SERVER | NO CODE | CLOUD HOSTED

Kevel MCP

Manage ad inventory and campaign structure by conversation.

Kevel connects your advertising operations directly to any AI client. This MCP manages ad campaigns, inventory sites, and creatives using natural language commands. You'll use it to list advertisers, check campaign configurations, pull zone details for specific sites, or audit creative assets without logging into the Kevel dashboard. It gives your agent hands-on control over complex ad serving processes.

A+ Quality Score 100/100

ad-serving | ad-ops | campaign-management | inventory-management | ad-tech | api-integration

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

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.

Kevel MCP

11 tools available
Cloud-hosted on Vinkius

Kevel lets you run your ad operations straight through your AI client. Instead of navigating multiple dashboards to track campaigns or check inventory health, you just ask your agent what you need. You can pull up detailed reports on campaign structures, query specific advertiser metadata, or list all available zones for a site ID. This MCP makes managing everything—from the initial campaign setup to the final ad creative audit—a conversation with your agent.

This level of granular control means product managers and ad operations teams get immediate answers without clicking through tabs. If you're building an automated workflow, Vinkius hosts Kevel, letting your AI client access these tools alongside thousands of others. You simply tell your agent to 'list all active flights for the summer sale,' and it executes the API call instantly.

Core Capabilities

01 — Audit ad creatives

You can get a full list and detailed status of every creative asset uploaded to the account.

02 — Manage campaign details

Retrieve specific information about campaigns, including their current configuration and related flights.

03 — Check site inventory zones

List all ad placement zones for a given website or channel to ensure proper ad delivery.

04 — Query advertiser data

Pull metadata and specific details about any advertising entity within the system.

05 — Inspect site placements

List all sites that are currently active in your ad serving network, checking for availability.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to the Kevel MCP and enter your required API key.

02 Connect this MCP to your preferred AI client (like Cursor or Claude).

03 Tell your agent exactly what you need—for example, 'List all zones for site X'—and it executes the request.

The bottom line is that once connected, your AI agent treats Kevel like a native function within its own command set.

Built For

Ad operations managers who hate manual dashboard checks. Product owners needing real-time inventory data without developer intervention. Ad tech engineers integrating ad serving logic into custom tools.

Ad Operations Manager

Uses the MCP to check campaign status and list flights across multiple campaigns on the fly, ensuring zero downtime.

Product Owner

Asks for site placements or zone details when designing a new page feature, getting instant confirmation instead of waiting for engineering.

Ad Tech Engineer

Retrieves technical metadata for ad creatives and supported ad types to validate integration specs before writing any code.

What Changes When You Connect

01 Audit creatives instantly: Use `list_creatives` to get a complete list of all your ads, so you never have to guess if an asset was uploaded correctly.

02 Control ad spend visibility: You can use `list_campaigns` and then `get_campaign` to pull detailed status reports on specific campaigns in seconds.

03 Verify placements easily: If you need to know what zones are available for a new client, running `list_zones` gives you the technical details right away.

04 Manage relationships quickly: Use `list_advertisers` or `get_advertiser` to pull metadata on specific companies without needing their account login credentials.

05 Check network availability: Running `list_sites` ensures that all your designated ad placements are online and ready for traffic.

Real-World Applications

A client asks about campaign performance across 10 channels.

Instead of logging into Kevel, the agent runs `list_campaigns` to get a list, then calls `list_flights` for each one. It compiles a single summary showing which campaigns are active and how many flights they have running.

I need to validate if our new website section supports native ads.

The agent first runs `list_ad_types` to confirm 'Native' is supported. Then, it uses `list_zones` and specifies the site ID to ensure there's an actual available ad placement zone for that format.

We suspect a major advertiser's profile data needs updating.

You ask your agent about 'Acme Corp.' The agent executes `get_advertiser` and pulls all the required metadata, which you can then review and pass to your internal CRM.

My team is onboarding a new site and needs to know its full inventory scope.

The agent calls `list_sites` for general coverage, followed by `list_channels` to understand the full range of distribution options available on that platform.

Patterns to Avoid

Treating Kevel like a simple database query

✗ AVOID

Telling your agent: 'Show me all ads for Summer Sale.' The tool needs more context to be useful.

✓ INSTEAD

Be specific. Use the tools by saying, 'First, run `list_campaigns` to find the ID for 'Summer Sale,' then use that ID with `get_campaign` to retrieve the full details.'

Ignoring required site IDs

✗ AVOID

Asking: 'What zones are available?' The tool fails because it doesn't know *where* you want the zones listed.

✓ INSTEAD

Always provide context. Use `list_zones` and include the specific Site ID (e.g., 12345) to get accurate placement data.

Trying to fix a campaign manually via chat

✗ AVOID

Saying: 'Change Campaign X budget to $5,000.' The MCP can only read and list data; it cannot write or modify settings.

✓ INSTEAD

Use this MCP to *read* the current state. Use `get_campaign` to confirm the existing budget before making manual changes in the dashboard.

The Right Fit

Use Kevel if your core workflow involves auditing, listing, or verifying ad serving infrastructure details. This MCP is perfect for Product Managers who need to validate inventory availability (`list_zones`) without logging into a dashboard, or Ad Tech Engineers needing granular metadata on creatives and formats using `list_ad_types`. Don't use this if you need to actively *change* campaign settings (like budget increases) or upload new assets; this MCP is purely for reading and retrieving data. If your goal is solely reporting performance metrics over time, you might need a dedicated analytics connector instead.

Ad Ops teams waste hours clicking through complex dashboards.

Today, checking ad inventory requires jumping between the campaign overview, the site listing, and the zone settings. You copy IDs from one screen, paste them into another to confirm if a creative asset matches a required format, then you manually check the status of all related flights. It's slow, error-prone work that takes you away from actual strategy.

With this MCP, your agent handles the clicks. Instead of juggling multiple tabs and copy-pasting IDs, you just ask it to 'List zones for site 12345.' The answer comes back in a clean format, giving you immediate data confirmation.

Get instant campaign visibility with Kevel.

The process of reviewing an ad's full context—from the advertiser to the specific creative type—used to mean running multiple reports and cross-referencing IDs. You'd have to check `list_advertisers`, then run `get_advertiser` for details, and finally call `list_creatives` just to confirm format compliance.

Now you can consolidate that entire audit into one conversation. Your agent pulls together all the necessary metadata from various endpoints in a single response, giving you total clarity without ever leaving your chat interface.

Kevel with 11 Tools

Use these tools to query campaign data, check available ad placements, retrieve advertiser details, and audit all creative assets through natural language commands.

#	TOOL	DESCRIPTION
01	`get_campaign`	Gets specific details about a single campaign you specify.
02	`list_zones`	Lists all the designated ad placement zones across your network.
03	`list_ad_types`	Retrieves a list of supported ad formats, like banner or native placements.
04	`list_ads`	Lists all the individual ads currently uploaded to your account.
05	`list_advertisers`	Provides a comprehensive list of every advertiser managed in Kevel.
06	`list_campaigns`	Lists all the active and archived campaigns currently running.
07	`list_channels`	Retrieves a list of all associated channels used for ad serving.
08	`list_creatives`	Lists every creative asset that has been uploaded to the account.
09	`get_advertiser`	Retrieves specific details for a single, named advertiser.
10	`list_flights`	Lists all ad flights associated with your campaigns.
11	`list_sites`	Lists every site that is part of the advertising network.

See It in Action

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

U Show me all active campaigns in Kevel.

I've retrieved your campaigns. You have 10 active campaigns, including 'Summer Sale 2024' and 'New Product Launch'. Would you like to see the flights for any of them?

U List all ad zones for the site with ID 12345.

I found 5 zones for site 12345: 'Header Banner', 'Sidebar Native', and 3 others. All zones are currently active.

U What ad types are supported in my Kevel account?

I've listed the supported ad types. Your account supports 15 formats, including 'Banner', 'Native', 'Video', and 'Interstitial'.

Frequently Asked Questions

01 **How do I find out what ad zones are available with Kevel?**

You use the `list_zones` tool. You must provide a specific site ID for the tool to list the correct and available placement zones on that website.

02 **Can I check which campaigns are running using Kevel MCP?**

Yes, you can use `list_campaigns` to get a full overview of all campaign IDs. After getting the ID, you run `get_campaign` for detailed status information.

03 **Does Kevel allow me to list ad creatives by type?**

You can use `list_creatives` to pull every asset uploaded. While it doesn't filter by type, you can then check the metadata of each creative to audit its format.

04 **What is the difference between `list_sites` and `list_channels` in Kevel?**

`list_sites` gives you a list of physical web properties (the placement location). `list_channels` provides a list of distribution methods or pathways for serving ads.

05 **How do I get details on a specific advertiser using Kevel MCP?**

You must use the `get_advertiser` tool and provide the unique identifier for that advertiser. This pulls all associated metadata in one go.

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 `"kevel": { "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

Kevel 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 Kevel. 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	Kevel MCP
Server ID	`019d75c0-7de4-71e3-a827-044efc1cdd4e`
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/kevel.