MCP SERVER | NO CODE | CLOUD HOSTED

api.video MCP for AI Agents

Manage live streams and analyze video performance metrics

api.video allows you to manage every aspect of video content—from initial upload and live stream setup to advanced analytics and chapter creation—all through natural conversation with your AI client.

A Quality Score 94.17/100

video-hosting | live-streaming | video-api | encoding | video-analytics

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/apivideo-alternative

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.

api.video MCP

49 tools available
Cloud-hosted on Vinkius

This MCP gives your agent total control over complex video infrastructure. You can handle the entire lifecycle, starting by ingesting raw footage from a URL or uploading it directly. It lets you manage the container itself, set up custom player themes, and even schedule watermarks for protection. For live content, you can create streams, manage broadcast details, and eventually signal when the session is complete. Beyond storage, your agent pulls detailed metrics like viewership breakdowns and time-series data to analyze performance. You'll also get deep control over metadata by creating summaries or listing chapters. If you're already using a platform like Vinkius, connecting this MCP gives your AI client all the video tools it needs in one place.

Core Capabilities

01 — Manage Video Assets

Create new video containers, upload sources from URLs, and update core parameters for any existing video.

02 — Control Live Broadcasts

Set up live streams, manage stream keys, and terminate running broadcasts via simple commands.

03 — Analyze Performance Data

Retrieve aggregated metrics or detailed time-series breakdowns to understand how videos are performing for your audience.

04 — Enrich Video Metadata and Presentation

Generate automatic summaries, list chapters, add captions, and customize the appearance of players with watermarks.

05 — Automate Content Pipelines

Set up webhooks to react instantly when video events occur and generate upload tokens for secure client-side data ingestion.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP in Vinkius and provide your api.video API Key.

02 Your AI agent accesses the full suite of video tools, allowing you to issue commands like 'List all videos' or 'Get metrics breakdown'.

03 The system executes the request against the live API connection, returning structured data that your agent translates back into actionable conversation.

The bottom line is that you talk to your AI client about video tasks—like analyzing performance or updating chapters—and it handles all the necessary backend calls automatically.

Built For

This MCP is essential for media companies, content marketing teams, and developers building visual platforms. It solves the problem of scattered video management by centralizing control over encoding, streaming, and analytics.

Video Editor

Needs to quickly update metadata or generate summaries for a batch of finished videos without logging into separate dashboards.

DevOps Engineer

Requires the ability to check encoding statuses, manage webhooks, and troubleshoot video containers directly from their code environment.

Content Manager

Manages a content library by listing videos, adding captions, or updating player themes across multiple projects.

What Changes When You Connect

01 Get instant content analysis: Use the `create_summary` tool to generate abstracts and key takeaways for any video, eliminating manual transcription work.

02 Maintain control over distribution: By running `create_player` and managing watermarks, you ensure your brand identity is consistently applied across all viewed videos.

03 Streamline live broadcasting: The MCP allows you to use `create_live_stream` and later `complete_live_stream`, handling the entire lifecycle of a broadcast session.

04 Deep analytics access: You don't just get numbers; you can call `get_metrics_breakdown` or `get_metrics_timeseries` to pull specific performance data points for reporting.

05 Simplify content ingestion: Instead of complicated API calls, your agent handles uploading sources using `upload_video_source` and managing secure tokens with `create_upload_token`.

Real-World Applications

Analyzing a Campaign's Reach

A marketing analyst needs to know which videos performed best last quarter. They ask their agent to use the MCP, which calls `get_metrics` and then runs `get_metrics_timeseries` to pull a detailed graph of viewership spikes over time.

Updating Evergreen Content

A content manager noticed an old video is missing key chapters. They ask their agent, which uses the MCP's tools to `list_chapters` first, then executes a specific chapter update using `get_chapter`, fixing the metadata without touching the source file.

Securing Upload Pipelines

A developer needs their client application to upload video files securely. They ask their agent to use `create_upload_token` and then monitor it with `get_upload_token`, ensuring the process is auditable and safe.

Handling Live Broadcasts

A producer needs to set up a multi-day training event. They ask their agent to use `create_live_stream` for the initial setup, and later confirm the end by calling `complete_live_stream`, fully managing the live cycle.

Patterns to Avoid

Manually updating player details

✗ AVOID

Logging into the web dashboard every time you need to change a brand color or logo on a video player.

✓ INSTEAD

Use the MCP's `update_player` tool. Your agent handles changing the theme and logo using `delete_player_logo` if necessary, keeping everything within your chat interface.

Forgetting to track content history

✗ AVOID

Deleting a video or making an update, only to realize later you needed that original version for comparison.

✓ INSTEAD

If Video Restore is active, use `list_discarded_videos` and then execute `restore_discarded_video` through the agent. Never lose historical context.

Mismanaging captions

✗ AVOID

Having to manually delete old or incorrect caption tracks one by one.

✓ INSTEAD

List all available captions using `list_captions`, then direct your agent to use `delete_caption` on the specific track that needs removing, keeping your metadata clean.

The Right Fit

Use this MCP if you need deep, automated control over a video's entire lifecycle: ingestion, streaming, metadata enrichment, and analytics retrieval. You should connect it when multiple steps are required—for instance, uploading content (`upload_video_source`) AND immediately generating summaries (`create_summary`). Don't use it if your only goal is simple viewing or basic hosting; you don't need the full power of this MCP for that. Instead, if you just need to list videos and nothing else, a simpler file management tool might suffice. But if you require reliable webhooks, player customization (`create_player`), or time-series performance data, this is your go-to video operations hub.

api.video MCP: Automating Video Content Management Workflows

Today, managing a single video asset means jumping between five different interfaces: the upload dashboard for sourcing footage, the analytics tab for checking views, the player settings page to change branding, and the metadata editor to write descriptions. This process is tedious, requires constant context-switching, and often leads to forgotten steps or inaccurate data entry.

With this MCP connected, your agent acts as a dedicated video engineer. You simply tell it what you need—like 'Generate summaries for all videos tagged Q4'—and it handles the uploads, calls `create_summary`, updates metadata, and reports back the finished product. It eliminates dashboard hopping entirely.

api.video MCP: Deep Dive into Live Streaming Operations

Setting up a live stream used to be a multi-step nightmare: first, creating the stream object; second, getting the correct keys; third, updating settings if you needed a different title or scheduled date. A simple error in any single step meant the broadcast failed.

Now, your agent manages this whole process conversationally. You can use `create_live_stream` and later modify it with `update_live_stream`—all without touching an API key or navigating complex forms. The entire workflow is managed through natural chat.

Video Stream Operations: 30 Tools for Content Management

Use these tools to list videos, create content containers, manage player themes, and retrieve detailed performance metrics.

#	TOOL	DESCRIPTION
01	`complete_live_stream`	Stops a live stream that is currently running.
02	`create_live_stream`	Creates a new object defining the parameters for a live broadcast.
03	`create_player`	Generates and customizes a player theme, allowing you to set specific visual elements.
04	`create_summary`	Automatically generates an abstract and key takeaways for a specified video content.
05	`create_upload_token`	Generates a temporary token used to securely upload large files from client-side applications.
06	`create_video`	Establishes a new video object container for tracking and management.
07	`create_webhook`	Subscribes to specific video events, ensuring you are notified when something happens to your content.
08	`delete_caption`	Removes a specific caption track from a video.
09	`delete_chapter`	Deletes defined time segment markers (chapters) from a video timeline.
10	`delete_live_stream`	Permanently removes an existing live stream definition.
11	`delete_player_logo`	Removes the logo image from a custom player theme.
12	`delete_player`	Deletes an entire customized player theme.
13	`delete_summary`	Permanently removes generated summary text and takeaways for a video.
14	`delete_upload_token`	Invalidates and deletes an active upload token.
15	`delete_video`	Permanently removes a video object from your project container.
16	`delete_watermark`	Removes a defined watermark overlay from the video content.
17	`delete_webhook`	Unsubscribes and removes an existing webhook listener.

#	TOOL	DESCRIPTION
18	get_caption	Retrieves the full text and timing data for a specific caption track.
19	get_chapter	Fetches the details and timecodes for a specific chapter marker in a video.
20	get_discarded_video	Retrieves historical data and details of a video that was discarded from the active project.
21	get_live_stream	Fetches all current settings and status information for a specific live stream.
22	get_metrics_breakdown	Pulls an analysis of viewership metrics, providing detailed breakdowns by source or time period.
23	get_metrics_timeseries	Retrieves performance data points over a specific timeline for trend analysis.
24	get_metrics	Gets an overall, aggregated view of all key analytics metrics for your content.
25	get_player	Retrieves the full configuration and theme details of a specific player type.
26	get_summary_source	Fetches the original text or source material used to generate a video summary.
27	get_upload_token	Retrieves the details and expiry time of an active upload token.
28	get_video_status	Checks if a video has finished uploading or encoding, providing its current readiness status.
29	get_video	Retrieves all core details about a specific video object in the project.
30	get_webhook	Fetches the configuration and security signature for an existing webhook listener.
31	list_captions	Lists all available caption tracks associated with a video container.
32	list_chapters	Retrieves a list of all defined chapters and their timecodes for a given video.
33	list_discarded_videos	Lists videos that were discarded, if the Video Restore feature is enabled on your account.
34	list_live_streams	Retrieves a list of all existing live stream definitions in the project.
35	list_players	Lists every customized player theme that has been created for your videos.

#	TOOL	DESCRIPTION
36	list_summaries	Retrieves a list of all generated summaries across your video library.
37	list_tags	Shows every tag used in the project and how many videos are associated with each one.
38	list_upload_tokens	Lists all active upload tokens to help manage your content pipeline security.
39	list_videos	Retrieves a list of every video object currently tracked in the project container.
40	list_watermarks	Lists all existing, defined watermarks used across your videos.
41	list_webhooks	Shows a list of all configured webhook endpoints and their current status.
42	restore_discarded_video	Moves a video from the discarded state back into active project management.
43	update_caption_default	Sets one specific caption track to be displayed by default on the player.
44	update_live_stream	Modifies existing live stream settings, such as titles or broadcast parameters.
45	update_player	Makes changes to the visual appearance and settings of an existing player theme.
46	update_summary_source	Allows manual editing or updating of the source text for a video summary.
47	update_thumbnail	Changes the preview image shown for a video by selecting a specific timecode.
48	update_video	Modifies general parameters of an entire video object, like its title or description.
49	upload_video_source	Ingests new video content directly from a provided web URL.

See It in Action

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

U List all videos I have, check which ones are ready to publish, and get their view count.

Video Inventory Report

VIDEO TITLE	ID	STATUS	VIEW COUNT
Product Demo Q1	vi4k0j...	Ready (100%)	8,902
Interview with Sarah	vi2l9m...	Encoding (720p)	1,550
Tutorial: Basics	viabcde...	Complete	22,101

Please review the details for 'Product Demo Q1' if you need to update its thumbnail.

U I need a new live stream for next week's conference. Can you set it up and give me the broadcast keys?

Live Stream Setup Complete

- ✅ **Stream Title:** Annual Tech Summit 2024
- ✅ **Start Date:** Oct 15, 2024
- ✅ **RTMP Endpoint:** rtmp://broadcast.api.video/s
- ✅ **Stream Key:** `7f8e9a...` (Keep this private)

You can start broadcasting anytime using the key provided. I've also updated the player theme to match your new branding.

U Can you analyze the performance of my last five videos and give me a summary?

Here is an aggregated view of the top 5 video performances:

- **Tutorial: Basics:** Highest engagement. Viewership peaked on Tuesday afternoon.
- **Product Demo Q1:** Strong retention rates, especially in the first 3 minutes.
- **Interview with Sarah:** Steady viewership, suggesting good organic sharing.

I've generated a full text summary for each one and updated their metadata so you can reference key topics easily.

Frequently Asked Questions

01 **How do I use the api.video MCP to manage my live broadcasts?**

You tell your agent to create a new stream, and it handles all the necessary setup, providing you with the required broadcast keys. It also lets you update settings or signal when the stream is finished.

02 **Can I use this MCP to analyze my video performance data?**

Yes. You can ask your agent to pull detailed analytics. It retrieves both overall metrics and time-series breakdowns, helping you pinpoint exactly when and why viewers dropped off.

03 **Is this better than manually uploading videos via the website?**

Much better. Instead of multiple web logins, your agent handles everything—from ingesting a video source from a URL to checking its final encoding status—all within one chat session.

04 **How do I add chapters or captions using the api.video MCP?**

You simply ask for it. Your agent uses tools to list existing chapters, and then you can direct it to create new ones at specific timecodes, updating your video's metadata instantly.

05 **Can I customize the player appearance with this MCP?**

Absolutely. You tell your agent what changes you want—like adding a watermark or changing colors—and it uses tools to update the player theme and logo without needing developer access.

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 `"apivideo-alternative": { "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

api.video 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 api.video. 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	api.video MCP
Server ID	`019e3865-87d2-72c7-832e-ab4ec5028474`
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/apivideo-alternative.