MCP SERVER | NO CODE | CLOUD HOSTED

OpenAI MCP

Build AI Workflows: Text, Images & Data Structuring

OpenAI MCP connects your AI client to the full suite of OpenAI tools, letting your agent perform advanced tasks like generating images (DALL-E 3), structuring complex data into reliable JSON, or converting text into searchable embeddings. It's a single connection that lets your workflow handle everything from creative media assets to deep content moderation and model fine-tuning.

A+ Quality Score 100/100

llm | generative-ai | embeddings | content-moderation | fine-tuning | image-generation

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

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.

OpenAI MCP

10 tools available
Cloud-hosted on Vinkius

Your AI agent can now access the core capabilities of OpenAI models directly through this MCP. Instead of needing multiple specialized APIs, you get a unified set of tools for handling complex data pipelines. You can ask your agent to generate responses using various GPT models; it can also create entirely new images from simple text prompts using DALL-E 3. For advanced data work, the connection handles converting raw text into vector embeddings, making semantic search reliable and fast. Need your output in a predictable format? The structured output tool ensures the response is perfect JSON every time. Plus, you can check content for policy violations or even run custom model training jobs. Because this entire catalog lives on Vinkius, connecting here gives your agent access to all these operations without switching services.

Core Capabilities

01 — Generate complex text and structured data

Your AI client can generate natural language responses using models like GPT-4o or force the output into a precise, predictable JSON format.

02 — Create media from descriptions

The system uses DALL-E 3 to produce images based solely on text prompts.

03 — Index and search unstructured text

It converts large amounts of raw text into vector embeddings, allowing your agent to perform semantic searches across massive knowledge bases.

04 — Monitor content policy compliance

The MCP checks any given piece of text against known policies for violations like hate speech or violence.

05 — Build custom, specialized models

You can manage and run fine-tuning jobs to create highly customized versions of the base models.

One Click on Vinkius — From Prompt to Execution

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

01 Your AI agent sends a request describing the required action—for example, 'Summarize this document and provide an image for it.'

02 The MCP routes that request to the appropriate tool, running both text generation and image creation in sequence.

03 You get back a single, cohesive result set: structured summary text alongside a direct URL link to the generated image.

The bottom line is your agent handles entire workflows—like drafting an article and generating accompanying graphics—in one conversation thread.

Built For

Content producers, data engineers building RAG systems, product managers designing AI features. If your job involves taking raw information and turning it into structured assets or media, you need this.

Technical Writer

Using the MCP, they can generate drafts, then immediately run content moderation checks on the text before submitting it for review.

Data Scientist

They use the embeddings tool to convert internal documents into searchable vectors, building robust knowledge bases that go beyond simple keyword matching.

Marketing Manager

Needs a new blog post. They ask their agent to write the copy and then immediately call the image generation tool for accompanying hero graphics.

What Changes When You Connect

01 Structured output ensures your agent never gives you messy text. You get reliable JSON data every time, perfect for feeding into databases or subsequent code blocks.

02 Need to search a huge internal document library? Running the `create_embedding` tool turns raw text into searchable vectors, making semantic searches possible—you find meaning, not just keywords.

03 Stop juggling asset tools. If you need an accompanying graphic for your content, the agent can run the `generate_image` tool right after writing the summary, giving you a complete package instantly.

04 Content compliance is key. Before publishing anything, use `moderate_content` to automatically check all text against policy guidelines and catch violations before they go live.

05 You're building specialized agents? The MCP handles fine-tuning jobs (`create_fine_tune`), letting you train custom models on proprietary data without leaving your primary workflow.

Real-World Applications

Drafting a marketing campaign with assets

A marketer asks their agent to draft three social media posts. The agent uses `chat_completion` for the copy and then calls `generate_image` three times, returning both text and associated visual URLs in one response.

Building a corporate knowledge search bot

An engineer uploads hundreds of PDF reports. The agent uses `create_embedding` to turn these PDFs into vectors. Later, when asked a question, the agent searches these vectors and summarizes the answer using `chat_completion`.

Pre-flight content review pipeline

A technical writer pastes a draft article. The agent first runs `moderate_content` to check for compliance, then uses `structured_output` to pull out key talking points into a structured report.

Creating a niche customer service bot

A product team trains a model using the `create_fine_tune` tool on 10k support tickets. The resulting custom assistant can then answer specific, highly technical questions via `list_assistants`.

Patterns to Avoid

Treating OpenAI as just a text generator

✗ AVOID

The user only prompts the agent to 'Write me an article.' The resulting text is good, but the workflow stalls because they have to switch tools manually to generate an image or structure data.

✓ INSTEAD

Tell your agent exactly what you need. Use `chat_completion` for the draft, then explicitly ask it to use the `generate_image` tool and wrap up the whole request with a call to `structured_output` for a final JSON summary.

Using embeddings only for search

✗ AVOID

The user treats embedding results like simple keywords. They get vector matches, but they don't know how to feed those back into the agent for contextual answers.

✓ INSTEAD

After running `create_embedding`, pass the retrieved context back into a final `chat_completion` call. This lets your AI client use the data to generate an actual, coherent answer, not just a list of sources.

Skipping content validation

✗ AVOID

A marketing team runs copy through their agent and publishes it without checking for policy issues. They get flagged later because of unmoderated text.

✓ INSTEAD

Make `moderate_content` the mandatory first step in your workflow. Always check the output before publishing to ensure clean, compliant content.

The Right Fit

Use this MCP if your task requires multiple, distinct OpenAI capabilities—for instance, generating text AND images, or creating embeddings AND structured JSON. It's built for complex orchestration. Don't use it if you *only* need basic chat responses; sometimes a simpler connector might suffice. However, if the core requirement is reliable data structure (e.g., 'I must get this output as

a list of objects'), then `structured_output` makes this MCP mandatory. If your goal is simply to retrieve information from an existing database, you only need embeddings and don't require the full power of chat completions or image generation.

The asset creation cycle feels like managing three different platforms.

Today, if you write a piece of content for the web, you usually open one tab for writing (the text), another for generating the main image assets, and sometimes a third tool just to check that your copy doesn't violate any guidelines. You spend time copying prompts, pasting URLs, and managing three separate API keys or logins.

With this MCP, you tell your agent what you want once. It handles everything: writing the body text using `chat_completion`, then calling `generate_image` for a hero shot, and finally running `moderate_content` to certify it all—and it hands you one cohesive result.

Structured JSON Output Guarantees Clean Data Every Time

Before this MCP, if an AI generated a summary list, the output was often natural language text with bullet points and varying formatting. You had to write custom code every time to parse that messy string just to get the key data fields.

Now, by using `structured_output`, you define exactly what the result should look like—a specific schema of JSON objects. Your agent spits out perfect, machine-readable data instantly.

OpenAI - 10 Tools Available

These tools let your agent handle everything from creating visual assets and running content checks to generating structured data outputs.

#	TOOL	DESCRIPTION
01	`create_fine_tune`	This tool initiates a specialized training job using your uploaded data file ID to customize model behavior.
02	`list_fine_tunes`	You can check the status and list all existing fine-tuning jobs you've started.
03	`chat_completion`	Generate conversational text responses by specifying a model like gpt-4o or gpt-4o-mini.
04	`structured_output`	Force the AI to generate output that strictly follows a defined JSON schema from your prompt.
05	`list_models`	Retrieves a list of all available OpenAI models you can use in your prompts.
06	`create_embedding`	Converts any piece of text into a dense numerical vector representation for indexing.
07	`generate_image`	Creates an image file and returns its direct URL based on your descriptive prompt using DALL-E 3.
08	`list_files`	Retrieves a list of files previously uploaded to the OpenAI system for training or reference.
09	`moderate_content`	Checks provided text against policy guidelines and returns a violation score for multiple categories.
10	`list_assistants`	Lists any configured OpenAI assistants that your agent can interact with or manage.

See It in Action

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

U Ask GPT-4o to summarize this document in 3 bullet points.

Here's the GPT-4o summary:

- **Revenue grew 23%** YoY driven by enterprise contracts
- **Operating costs decreased** 8% through automation initiatives
- **Customer retention** reached 94%, highest in company history

U Generate an image of a futuristic cityscape at sunset.

🎨 Image generated!

- **URL**: https://oaidalleapiprodscus.blob.core.windows.net/...
- **Size**: 1024×1024
- **Revised prompt**: A breathtaking futuristic cityscape...

U Check if this text violates content policies.

✅ Content analysis complete:

CATEGORY	FLAGGED	SCORE
Hate	No	0.001
Violence	No	0.002
Sexual	No	0.000
Self-harm	No	0.000

No policy violations detected.

Frequently Asked Questions

01 **How does the OpenAI MCP handle images?**

You use the `generate_image` tool to create pictures using DALL-E 3. You simply provide a text description, and it returns a direct URL for the generated asset.

02 **Is the content moderation tool reliable?**

The `moderate_content` tool checks text against established policies for violations like hate speech or violence, giving you scores and clear flags on compliance status.

03 **Can I train a custom model with this MCP?**

Yes. You manage the process using `create_fine_tune` to upload data and start training jobs, and then check progress via `list_fine_tunes`.

04 **Does the OpenAI MCP support multiple AI models?**

The chat completion tool allows you to specify various models, such as gpt-4o or gpt-4o-mini, letting you pick the right model for the job.

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 `"openai": { "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

OpenAI 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 OpenAI. 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	OpenAI MCP
Server ID	`019d75e8-7403-71b1-8a02-f63f21a4c9a1`
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/openai.