MCP SERVER | NO CODE | CLOUD HOSTED

Fireworks AI MCP

Build complex generative tasks in chat.

Fireworks AI gives your agent ultra-fast access to advanced generative models for everything from chat conversations to image creation. It lets you synthesize embeddings, transcribe audio files, or generate text completions instantly, all through one single connection point.

A+ Quality Score 100/100

llm-inference | generative-ai | embeddings | model-deployment | high-performance-api | ai-orchestration

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

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.

Fireworks AI MCP

6 tools available
Cloud-hosted on Vinkius

This MCP connects your favorite AI client directly to Fireworks AI's high-speed model infrastructure. You get full control over running generative inference without needing complex setups. Need to build a semantic search tool? Use the embeddings synthesis capability. Want to create marketing visuals on the fly? Generate them from text prompts. The connection also lets you transcribe audio files or run chat completions against optimized LLMs.

It's designed for developers who need speed and reliability in their AI workflows, letting your agent talk to multiple specialized services through one place. This simplifies integration dramatically; instead of managing several separate API keys, you connect once via Vinkius and get access to all these high-performance tools.

Core Capabilities

01 — Run Chat Conversations

Your agent can send chat messages and receive responses from ultra-fast LLMs hosted by Fireworks AI.

02 — Create Vector Embeddings

Generate multi-dimensional vector representations for any array of text strings, making them ready for semantic search or indexing.

03 — Synthesize Images from Text

Command the system to generate high-fidelity images using descriptive text prompts.

04 — Transcribe Audio Files

Pass a public URL for an audio file and receive a flawless, structured textual transcription.

05 — Generate Text Continuations

Complete instructions or prompts by generating basic, high-quality text continuations using state-of-the-art models.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and input your Fireworks AI API Key into the Vinkius catalog.

02 Your AI client detects the available tools, allowing you to call functions like `embed` or `image` using natural conversation.

03 The system sends the request to the Fireworks backend, returning the generated data—be it a vector array or a transcribed text string.

The bottom line is that you get fast access to multiple specialized generative AI services through your existing chat interface.

Built For

Engineers and data scientists who hate slow, complex API calls need this. If your workflow involves turning text into searchable vectors or generating media assets on demand, this is for you.

AI Developer

You use it to test and debug LLM prompts and inference parameters against real-world models without writing boilerplate API integration code.

Data Scientist

You quickly generate embeddings for document sets, then run `list_models` to ensure you're using the most efficient model for your RAG pipeline.

Product Manager

You test generative features in natural language conversation to validate if the AI can handle edge cases before handing off code to engineers.

What Changes When You Connect

01 Generate searchable vectors with `embed`. You can feed it a list of sentences and get back the multi-dimensional arrays needed for semantic search, skipping manual vector library calls.

02 Need visuals? Use the `image` tool to create high-fidelity pictures directly from text prompts. It's perfect for rapidly prototyping assets when you don't have design time.

03 The `transcribe` function lets your agent pull structured text out of any audio file by passing just a public URL, making media processing simple.

04 `chat` handles the heavy lifting of conversation orchestration against ultra-fast LLMs. Your agent keeps track of context across multiple turns without you having to manage session state.

05 Before building anything, use `list_models`. This tool lets you check what high-speed models are available and get their specific IDs so your project stays up-to-date.

Real-World Applications

Processing a Meeting Recording

A product manager uploads an audio recording from a client meeting. They ask their agent to transcribe it using `transcribe`. The resulting text is then passed back into the chat tool, asking the agent to summarize action items and identify key pain points.

Building a Document Index

A data scientist has thousands of product manuals. Instead of writing complex code for every document, they ask their agent to run `embed` on chunks of text from the manuals. This instantly provides the vector arrays needed to index the knowledge base.

Creating Marketing Content

A marketing team needs a hero image for an upcoming campaign. They prompt their agent, 'Generate a cyberpunk city at sunset.' The `image` tool runs the inference and returns the visual asset immediately for review.

Debugging LLM Prompts

An AI developer wants to see how different models handle complex instructions. They use the `chat` tool, cycling through multiple model IDs retrieved via `list_models`, to compare outputs quickly and debug their prompt logic.

Patterns to Avoid

Assuming Model Availability

✗ AVOID

A developer tries to run a chat function using an old or unverified model name, resulting in an 'Model Not Found' error and stalling development.

✓ INSTEAD

Always start by running `list_models`. This guarantees you have the current list of available IDs for high-speed inference, making your code resilient to updates.

Overcomplicating Content Creation

✗ AVOID

A user tries to manually stitch together embedding generation, image creation, and text completion using three different API clients.

✓ INSTEAD

Use this MCP. You can manage all these tasks—embeddings via `embed`, images via `image`, and chat completions via `chat`—all from one natural conversation flow.

Ignoring Input Validation

✗ AVOID

Sending an audio URL to the agent without checking if it's publicly accessible, causing the transcription tool to fail immediately.

✓ INSTEAD

Before calling `transcribe`, verify the public accessibility of your source material. The tool requires a public URL to function correctly.

The Right Fit

Use this MCP if your core task involves combining multiple types of generative AI operations in one pipeline: text conversation, media creation, and data vectorization. You need fast inference that can handle everything from `chat` sessions to image synthesis (`image`) without switching tools or APIs.

Don't use it if you only need a single function, like simple keyword lookups in a database (use a dedicated database connector) or if your task is purely offline data processing. If you just need to generate text completions for basic forms, the `completion` tool works well, but if you need semantic search on that content, you'll also want `embed` . This MCP shines when you have multi-step workflows spanning different media types.

Manually handling diverse data inputs is a constant headache.

Think about the process today. You get an audio file, so you copy it into a transcription service and wait for text to populate. Then, you have that text and need to summarize it in Notion, which requires another copy-paste cycle. If you suddenly realize you also needed vector embeddings of that transcript for your search index, you're staring at yet another dashboard and API key.

With this MCP, the flow changes completely. You hand the audio file over to your agent, and it handles the transcription using `transcribe`. Once that text is ready, you can immediately ask it to summarize the action items *and* simultaneously use the generated text to run `embed` for indexing—all in one conversation.

Generate Media & Embeddings with Fireworks AI

The biggest manual time sink is the handoff between media types. You generate an image using a separate service, then you copy that image description into your chatbot to get metadata, and finally, you have to feed all those strings back into a vector store's dedicated API.

Now, you can ask your agent to do it all in one go. Prompt for the visual asset using `image`, and immediately follow up with a request to run `embed` on the prompt description itself. The whole pipeline happens inside your chat window.

Fireworks AI with 6 Tools

Use these tools to manage your entire generative workflow—from creating visual assets and transcribing recordings to generating semantic vector data.

#	TOOL	DESCRIPTION
01	`embed`	Generates vector embeddings for a given set of text strings using Fireworks AI.
02	`list_models`	Retrieves an enumerated list of all available high-speed models hosted by Fireworks AI.
03	`image`	Creates a new, high-fidelity image based on the text description you provide.
04	`chat`	Engages in a multi-turn chat conversation with Fireworks AI's optimized language models.
05	`completion`	Generates basic textual completions for continuing an existing prompt or instruction.
06	`transcribe`	Processes a public URL to transcribe the audio content contained within that file.

See It in Action

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

U Chat with 'llama-v3-70b': 'Explain quantum entanglement simply.'

Inference complete! Llama-v3 response: 'Quantum entanglement is a phenomenon where two or more particles become connected in such a way that the state of one particle instantly influences the state of the other, regardless of the distance between them...'

U Generate embeddings for these sentences: ['AI is great', 'MCP is powerful']

Embeddings synthesized! I've retrieved the vector representations for your 2 sentences. You can now use these arrays for semantic search or indexing in your vector database.

U Generate an image of a cybernetic forest at night

Image generation started! I'm using Fireworks AI inference to create your cybernetic forest visual. The high-fidelity result will be ready for you to view in just a few seconds.

Frequently Asked Questions

01 **How fast is the model inference when I use Fireworks AI MCP?**

The core benefit of this MCP is speed. It connects you to ultra-fast LLMs, meaning complex tasks like `chat` completions or text generation happen much quicker than with standard API connections.

02 **Can I generate images using the Fireworks AI MCP?**

Yes, you can use the dedicated `image` tool. Simply provide a text prompt—like 'a neon jungle at night'—and the system returns a high-fidelity visual asset.

03 **What is the difference between `chat` and `completion`?**

The `chat` function is designed for multi-turn conversations, remembering context across several messages. The `completion` tool is better suited when you just need to finish a single instruction or prompt continuation.

04 **Do I need special setup for audio transcription with Fireworks AI MCP?**

No. You only need to provide the public URL of the audio file when calling `transcribe`. The tool handles the processing and returns clean, structured text.

05 **How do I know which models are available before using chat?**

You should use the `list_models` tool first. This enumerates all active model IDs and versions, letting you pick exactly what you need for your inference.

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 `"fireworks-ai": { "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

Fireworks AI 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 Fireworks AI. 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	Fireworks AI MCP
Server ID	`019d759a-23db-713a-b7ee-fa212fbba5a9`
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/fireworks-ai.