MCP SERVER | NO CODE | CLOUD HOSTED

Apiary MCP for AI Agents

Define, test, and publish robust REST API specifications

Apiary lets your AI agent manage the entire API life cycle—from initial design blueprint to published documentation and rigorous testing. You can read raw API specifications, update documentation instantly with new markdown code, or run Dredd-style tests against live backends, all without leaving your editor.

F Quality Score 14.04/100

api-design | swagger | documentation | api-blueprints | rest-api | workflow-automation

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

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.

Apiary MCP

10 tools available
Cloud-hosted on Vinkius

Managing APIs used to mean constantly jumping between tools: the design spec file, the documentation portal, and the actual test runner. Now, you can keep your whole process inside a single conversation with your AI agent. This MCP connects directly to your Apiary workspace, letting you take full command of API design and validation through natural language commands.

Whether you're an engineer needing to fetch the current blueprint or a writer updating documentation, your agent handles it. You can instruct it to publish changes, review team projects, and validate endpoints against existing specifications. It’s like having the entire API governance console open right in your chat window. This integration is hosted on Vinkius, giving you access to thousands of other development tools alongside Apiary.

Core Capabilities

01 — List all available APIs

Retrieves a comprehensive list of every API project configured within your connected Apiary workspace.

02 — Fetch raw API blueprints

Grabs the full source code for any specific API blueprint, whether it's in Markdown or Swagger format.

03 — Update and publish documentation

Publishes changes to an existing API project by accepting new markdown content, automatically updating both mock servers and public docs.

04 — Validate APIs with integrated testing

Executes Dredd-style tests against a specified API blueprint to ensure your backend implementation matches the design specifications.

05 — Review team structure and members

Manages team permissions by listing all teams you belong to, viewing team details, and seeing every member associated with that group.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP on Vinkius and provide your personal Apiary token.

02 Give your AI client a clear instruction, like 'Check the blueprint for X API' or 'Publish these docs'.

03 The agent interacts with the required tools, fetching raw blueprints, running tests, or pushing updates back to your workspace.

The bottom line is you get full control over your API design and documentation workflow without ever needing to copy anything out of your chat window.

Built For

This MCP is for technical teams who struggle with API specification drift—the gap between documented plans and actual code. If you're an engineer wasting time switching between documentation editors, or a manager needing to track team API ownership, this connector solves that pain point.

Backend Engineer

Needs their agent to fetch the current blueprint, write new endpoint specifications into the markdown, and then publish those changes back automatically.

Technical Writer

Uses the MCP to review and fix API documentation formatting on a remote server and submit corrected sections directly for publishing.

Engineering Manager

Needs visibility into team ownership, using it to list all active APIs under the department or check member permissions across different teams.

What Changes When You Connect

01 Keep documentation synchronized with reality. You can use `publish_blueprint` to update your live project specs directly from a chat prompt, eliminating manual copy-pasting.

02 Accelerate validation cycles by letting your agent execute `run_tests`, validating backends against the defined blueprint without switching tools.

03 Gain full visibility into team ownership. Use the team management tools like `list_team_members` to quickly see who owns which APIs and manage permissions.

04 Never lose track of specs again. With `get_api`, you can instantly fetch raw Markdown or Swagger code for any project, making it easy to share with teammates.

05 Instantly know where your API lives. The `get_doc_url` tool provides the live documentation and mock server URLs in one shot.

Real-World Applications

A new endpoint needs defining

The backend engineer asks their agent to fetch the current blueprint using `get_api`, writes a few lines of Markdown for the new endpoint, and then tells the agent to `publish_blueprint`. The change goes live immediately.

Checking team compliance

The engineering manager asks their agent to use `list_team_apis` for a specific department. This instantly shows all relevant projects, allowing them to check ownership and ensure proper coverage across the board.

Validating production changes

Before deploying code, the developer instructs their agent to `run_tests`. The MCP executes Dredd-style tests against the blueprint, providing a detailed pass/fail report instantly within the chat.

Updating external documentation

The technical writer needs to fix formatting on an old API. They use `get_api` to pull the code, correct it in their editor, and then ask the agent to `publish_blueprint`, pushing the corrected version live.

Patterns to Avoid

Manual spec drift

✗ AVOID

A developer updates the API documentation on the public site but forgets to update the original Markdown blueprint file, causing client code to break.

✓ INSTEAD

Always use your agent to manage the entire cycle. First, fetch the current specs with `get_api`, make changes, and then push them live using `publish_blueprint`.

Skipping team checks

✗ AVOID

An engineer assumes they have permission to modify an API because it works locally, but it turns out the team structure prevents publishing.

✓ INSTEAD

Before making changes, use `list_teams` and then `list_team_members`. This confirms both your access rights and who else needs to approve the change.

Ignoring test results

✗ AVOID

A feature is deployed because it 'looks right,' but critical edge cases break in production, leading to downtime.

✓ INSTEAD

Never skip validation. Use `run_tests` with your agent on the target project. The resulting report guarantees that the implementation matches the defined contract.

The Right Fit

Use this MCP if your workflow involves constantly moving API specifications between design, documentation, and testing environments. You need to ensure the blueprint (the design) perfectly matches the code (the reality). Don't use it if you only need simple data retrieval or general chat help; for that, a basic document search tool is fine. However, if your goal is merely listing projects without any write access, then `list_apis` alone might be enough. But if you have to *update* anything—the docs, the specs, or run validation against them—you need this full MCP. It gives you the write and validate permissions.

Apiary: Managing API Blueprints with AI Agents

Right now, updating an API spec is a miserable process. You're stuck jumping between your documentation tool to check formatting, the separate blueprint editor to make changes, and then manually copying that finished code into your version control system. It’s slow, tedious, and easy to get something wrong.

With this MCP, you just talk to your agent. Tell it what needs fixing or updating. The agent handles fetching the raw content, applying your edits, and publishing the finalized specs directly back into your workspace. You stay in one place.

Apiary: Automating API Team Governance with AI Agents

Before this MCP, figuring out who owned a specific API was a headache. You'd have to log into the team dashboard, navigate through multiple folders, and manually check membership lists just to confirm permissions.

Now, you simply ask your agent to list all APIs belonging to a certain team or check which members are associated with a project. It pulls that data instantly, giving you immediate oversight of your entire engineering ecosystem.

10 Tools for Managing API Blueprints and Testing

Use these tools to list projects, fetch blueprints, run tests, manage team members, and update documentation within Apiary.

#	TOOL	DESCRIPTION
01	`list_apis`	Lists every API project available within your connected Apiary workspace.
02	`get_api`	Retrieves the complete, raw source code blueprint for any specific API project.
03	`publish_blueprint`	Updates an existing API project's blueprint using new Markdown or Swagger specification content.
04	`get_me`	Fetches and displays your own authenticated user profile information from Apiary.
05	`list_team_apis`	Shows all API projects that belong to a specific team within the workspace.
06	`list_teams`	Lists every team you are currently a member of in Apiary.
07	`get_team`	Provides detailed information about a specific API team, including its purpose and owners.
08	`list_team_members`	Lists every user belonging to an Apiary team, showing their roles within that group.
09	`run_tests`	Runs Dredd-style API tests against a project blueprint and reports the validation status.
10	`get_doc_url`	Provides the live URLs for both the documentation page and the mock server associated with an API project.

See It in Action

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

U Hey, can you give me the current blueprint code for the payments API? I want to see if it still supports refund endpoints.

API Blueprint: Payments Gateway

- **Endpoints:** /payments (POST), /refunds (GET)
- **Status:** Active. Last Updated: 2024-10-05.
- **Details:** The refund endpoint requires a `transaction_id` and accepts an optional `reason` string.

```markdown
# Payments API
... (Full Swagger/Markdown code here) ...
```

U I wrote some new documentation for the user service endpoints. Can you publish this markdown update to the 'users-api' project?

✅ **Success:** The updated Markdown has been successfully published to `users-api`.

- **Validation:** Syntax verified as valid Blueprint.
- **Impact:** Mock server and public documentation URL refreshed automatically.

The new changes are now live for testing.

U Can you check the status of the user management API? I think there might be a bug in the authentication flow.

Dredd Test Report: User Management API

TEST CASE	STATUS	DETAILS
`list_users` (GET)	✅ PASS	Returns 200 OK. Correct schema verified.
`create_user` (POST)	❌ FAIL	Expected HTTP 401, but received 200 OK. Missing required 'role' field.

Frequently Asked Questions

01 **How does the Apiary MCP help me with API documentation updates?**

It lets you publish changes directly from your conversation window. You provide new markdown code, and the agent publishes it to the live project, automatically updating both the mock server and the public documentation URL.

02 **Can I use this MCP to test my API endpoints?**

Yes, you can run Dredd-style tests. Simply ask your agent to run tests against a specific blueprint. It executes validation checks and gives you an immediate pass/fail report on the backend implementation.

03 **Does the Apiary MCP help me manage my API team?**

Absolutely. You can list all APIs owned by your entire department or check who exactly is a member of a specific development team, helping you keep track of ownership and permissions.

04 **What if I need the raw code for an existing API blueprint?**

You can retrieve the full source code for any API project using the `get_api` tool. This gives you a clean, raw Markdown or Swagger file to work with in your own editor.

05 **Is this MCP only for single APIs?**

No, it can manage entire portfolios. You can list all available APIs across your whole workspace and even see which projects belong to a specific team.

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 `"apiary": { "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

Apiary 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 Apiary. 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	Apiary MCP
Server ID	`019d754f-13d5-7043-9413-ab75ec022ca0`
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/apiary.