MCP SERVER | NO CODE | CLOUD HOSTED

openFDA MCP

Cross-reference drug safety, food recalls, and device failures.

openFDA: Access comprehensive U.S. public health records, including drug adverse events, food recalls, and medical device safety reports directly through your AI client. This MCP lets you construct complex queries using real-world data from the FDA databases.

A+ Quality Score 100/100

pharmacovigilance | medical-devices | food-safety | regulatory-data | public-health

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

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.

openFDA MCP

3 tools available
Cloud-hosted on Vinkius

You're dealing with critical public health information that lives across three massive regulatory datasets: drugs, food items, and medical devices. Instead of manually logging into multiple government websites and wrestling with complicated API key setups, this MCP lets your agent query all three areas in one go. You can ask highly specific questions—for example, finding every reported adverse event linked to Ibuprofen *and* tracking active Salmonella recalls in California—using raw Lucene syntax. The result is structured data pulled directly into your workflow. It's a massive time-saver for anyone doing deep research or compliance checks. Because Vinkius hosts this MCP, you connect once from any compatible client and gain access to all these critical public health records.

Core Capabilities

01 — Investigate Drug Side Effects

Query millions of patient reports detailing documented side effects, medication errors, or product quality complaints.

02 — Track Foodborne Outbreaks

Search historical and current FDA enforcement reports for pathogens like Salmonella or Listeria.

03 — Monitor Medical Device Safety

Find records of malfunctions, injuries, or deaths associated with specific medical hardware.

04 — Perform Advanced Data Queries

Run multi-variable analytical research by inputting raw query syntax across all three databases.

One Click on Vinkius — From Prompt to Execution

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

01 You define the exact parameters for your search, specifying criteria like drug names, adverse reactions, or geographical locations.

02 Your AI agent uses the MCP to translate those parameters into a complex query understood by the FDA databases.

03 The system returns structured public health records, allowing you to analyze trends and specific incidents immediately.

The bottom line is that your agent pulls highly technical, real-time regulatory data without needing dozens of API keys or manual web scraping.

Built For

This MCP is for the compliance officer who can't afford to miss a single safety report, and the researcher drowning in raw public health documentation. If your job requires deep data correlation between drug usage, food supply chains, and medical hardware failures, this is essential.

Pharmacovigilance Specialist

Runs queries to check for links between specific medications and newly reported adverse events across millions of records.

Food Safety Analyst

Tracks outbreaks, searching the openFDA Food Enforcement database for pathogens like Salmonella or Listeria in specific regions.

Regulatory Compliance Officer

Validates company product safety by checking medical device records and ensuring compliance with FDA reporting standards.

What Changes When You Connect

01 Instead of sifting through scattered government PDFs, you can query all three databases—drug events, food recalls, and medical devices—in a single command. This saves hours of manual data aggregation.

02 The openFDA MCP lets your agent use raw Lucene syntax for maximum control. You aren't limited to predefined search fields; you define the exact variables needed for analysis.

03 You can rapidly check medication safety by using query_drug_events, finding adverse event patterns related to specific drugs or reactions across millions of records.

04 Compliance checks become simple: Use query_food_recalls to track if a pathogen like Salmonella was reported in a specific state and what the recall status is.

05 Monitor hardware integrity with query_medical_devices. This lets you check for malfunctions tied to generic device names or date ranges, which is crucial for risk assessment.

Real-World Applications

Investigating a new drug side effect

A researcher needs to know if Ibuprofen has any reported links to insomnia. They ask their agent to run a query using the 'query_drug_events' tool, getting immediate confirmation of multiple adverse event records without leaving their workspace.

Auditing a food supply chain breach

A public safety team suspects a contaminated product. They use 'query_food_recalls' to track all Salmonella outbreaks in California, allowing them to pinpoint the exact source and recall status immediately.

Analyzing device failure trends

A manufacturer needs to know about potential issues with pacemakers. They run 'query_medical_devices', filtering by malfunction event type and date range, giving them actionable data for a safety bulletin.

Patterns to Avoid

Searching only one database

✗ AVOID

Trying to determine if an adverse drug event is linked to a recalled food item. This requires checking two different systems manually.

✓ INSTEAD

Use the openFDA MCP to combine these searches by querying both 'query_drug_events' and 'query_food_recalls' in one workflow, correlating the data points instantly.

Using vague search terms

✗ AVOID

Asking a general question like 'Is anything wrong with drugs?' which yields thousands of unhelpful results.

✓ INSTEAD

Use specific tools and syntax. For example, use query_drug_events to search for `patient.drug.medicinalproduct:"ASPIRIN"` combined with a reaction type.

Ignoring date limitations

✗ AVOID

Running a device safety check without specifying a timeframe, resulting in an unmanageable volume of old data.

✓ INSTEAD

Always include specific date ranges when using query_medical_devices. Define the `date_of_event: [YYYYMMDD TO YYYYMMDD]` to narrow your focus.

The Right Fit

Use this MCP if your job requires cross-domain correlation of public health safety data. This is for deep dives involving drug side effects, food contamination reports, and medical hardware failures. Don't use it if you simply need general information; the depth of the available tools—like query_drug_events—requires precise knowledge (e.g., Lucene syntax) to get accurate results. Conversely, don't use this MCP for internal corporate data like HR records or sales figures; these databases are strictly public health regulatory archives.

Gathering Safety Data Used To Be a Database Nightmare

Before connecting openFDA to your agent, you had to manage three separate login portals. You'd run reports on drug adverse events in one place, then switch over to check for food recalls related to Salmonella in another. If you needed to correlate a medication failure with a device malfunction, you were copy-pasting criteria between three different spreadsheet tabs.

Now, your agent handles the complexity. By using this MCP through Vinkius, you define the scope of the problem—say, 'all adverse events related to pacemakers and specific drugs'—and get one cohesive data pull from all regulatory sources. You spend time analyzing insights, not managing logins.

Querying Public Health Records with openFDA

The manual steps that disappear are the repetitive searches for specific pathogens across state lines and the tedious process of cross-referencing device names against known adverse event dates. You don't have to switch between query_food_recalls and query_medical_devices just to build a timeline.

What’s different now is speed and scope. Your agent doesn't just search; it constructs complex, multi-variable queries across the entire U.S. regulatory record instantly.

openFDA: 3 Tools for Regulatory Data

These tools let you programmatically search three distinct U.S. public health databases to correlate adverse events, recalls, and device failures.

#	TOOL	DESCRIPTION
01	`query_drug_events`	Searches the Drug Adverse Events database for reports of side effects, medication errors, or product quality complaints using Lucene syntax.
02	`query_food_recalls`	Finds active and historical foodborne illness outbreaks and FDA enforcement reports by reason, status, or state.
03	`query_medical_devices`	Searches the Medical Device Adverse Events database for records of malfunctions, injuries, or deaths associated with medical hardware.

See It in Action

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

U What are the most recent food recalls related to Salmonella in California?

I've searched the openFDA Food Enforcement database using the Lucene query `reason_for_recall:"salmonella" AND state:"CA"`. Here are the most recent recall events and associated product descriptions.

U Are there any reports of 'insomnia' after taking generic Ibuprofen?

Searching the Drug Adverse Events dataset for `patient.drug.medicinalproduct:"IBUPROFEN" AND patient.reaction.reactionmeddrapt:"INSOMNIA"`. Yes, I've found multiple adverse event records. Here is a summary of the metadata.

Frequently Asked Questions

01 **How do I find reports on specific medication side effects using openFDA?**

You use the 'query_drug_events' tool. You input criteria like `patient.drug.medicinalproduct:"ASPIRIN"` combined with the desired reaction to pull adverse event records.

02 **Does openFDA allow me to query food recalls by state?**

Yes, you can use 'query_food_recalls' and include the state parameter (e.g., `state:"CA"`) alongside the reason for recall to narrow your search.

03 **What kind of data does openFDA provide on medical devices?**

The 'query_medical_devices' tool provides details on injuries, malfunctions, and deaths linked to specific device types or event dates. It tracks safety reports from the MAUDE database.

04 **Can I combine drug and food data in openFDA?**

Yes, you can build complex queries that reference both drug events and food recalls simultaneously, allowing for highly correlated public health research.

05 **Is the data from openFDA real-time or historical?**

The MCP accesses live regulatory databases, providing both historical records (like past outbreaks) and current, ongoing reports. Always check the database's update frequency for the absolute latest information.

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 `"openfda": { "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

openFDA 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 openFDA. 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	openFDA MCP
Server ID	`019d75e9-50a9-7053-bf2e-09a365f66fea`
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/openfda.