

M C P S E R V E R

N O C O D E

C L O U D H O S T E D

NYC 311 Service Requests MCP

Pull live New York City service data directly into your agent.

NYC 311 Service Requests MCP gives your AI client direct access to over 22 million service requests and call center inquiries from NYC Open Data. You can filter by borough, complaint type, or date to find specific pothole reports, noise complaints, or streetlight issues. It also includes a historic dataset covering 2010 to 2019 and detailed call center inquiry logs.

A+ Quality Score 96.67/100

new-york

nyc

311

service-requests

complaints

government-public-data



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.

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.

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.

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

Encryption at rest

Ed25519

PKI vault signatures

24h TTL

Ephemeral session keys

V8 Isolate

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.

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.

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.

06 — Honeypot Trap System

Phantom credentials are injected into isolated environments. If a honeypot 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.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

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.

NYC 311 Service Requests MCP

5 tools available

Cloud-hosted on Vinkius

You can now give your AI agent a direct line to New York City's non-emergency service data. This MCP connects your client to the massive NYC Open Data repository, covering everything from rodent complaints and water leaks to code enforcement and street conditions. Whether you are looking for recent trends or deep historical context, the data is right there.

You can use it to track how specific agencies like the DOT or NYPD are handling neighborhood issues. If you need to know why a certain area is seeing a spike in noise complaints, you can filter by borough or date to see the exact volume. It even handles the call center side of things, letting you see how information requests were resolved. Instead of manually digging through web portals, you just ask your agent to find the data for you.

Core Capabilities

01 — Trend Analysis

Your agent uses top complaint data to identify emerging city-wide issues.

02 — Specific Record Retrieval

Your agent pulls full details for a single complaint using its unique ID.

03 — Volume Calculation

Your agent calculates the exact number of matching complaints for any filtered criteria.

04 — Historical Lookups

Your agent accesses records from the 2010 to 2019 period when searching for specific keys.

05 — Call Center Auditing

Your agent examines how call center inquiries and information requests were handled.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/en/ai-agent-connect/nyc-311-service-requests — connect your AI agent in three steps.

- 01
- Connect your MCP-compatible client to Vinkius.
- 02
- Select the NYC 311 Service Requests MCP from the catalog.
- 03
- Ask your agent a question about NYC service requests or complaints.
- 04
- The agent uses the appropriate tool to query the NYC Open Data sets.
- 05
- The agent receives the raw data and presents the answer to you.

Connecting to this MCP gives your AI client immediate access to NYC's live service datasets.

Built For

This MCP is built for anyone needing to query massive amounts of NYC civic data without manual searching.

Newsroom Researchers

They hand off data verification and trend spotting to their agent.

Civic Tech Developers

They use the tool to feed real-time city data into custom applications.

Local Government Analysts

They use the agent to quickly quantify neighborhood service needs.

What Changes When You Connect

- 01
- Accesses over 22 million records without needing an API key.
- 02
- Automates the switch between current and historic datasets.
- 03
- Provides direct access to agency-specific complaint codes.

- 04 Eliminates manual filtering through the NYC Open Data web interface.
-

Real-World Applications

Neighborhood Issue Tracking

Check how many noise complaints have been filed in Brooklyn over the last month.

Agency Performance Review

Compare the volume of requests handled by the DOT versus the DEP.

Journalistic Fact Checking

Verify if a reported increase in rodent sightings is backed by 311 data.

Civic Planning

Identify high-traffic complaint areas to inform local infrastructure discussions.

Patterns to Avoid

Guessing request counts

❌ AVOID

You ask your agent to estimate how many pothole complaints happened in Brooklyn last month.

✓ INSTEAD

Use ``count_311_requests`` to get the exact number of requests matching your specific filters instead of guessing.

Searching only recent data

❌ AVOID

You try to find a service request from 2015 using a tool that only looks at current records.

✓ INSTEAD

Use ``get_311_request`` because it automatically checks current records and falls back to the 2010 to 2019 historic dataset if needed.

Ignoring call center logs

❌ AVOID

You assume all city interactions are logged as service requests and miss how staff handled general questions.

✓ INSTEAD

Use `search\_311\_inquiries` to access the call center dataset for information requests and resident transfers.

The Right Fit

Connect this MCP if you need to query NYC's public service datasets without managing API keys or massive spreadsheets. Do not connect it if you only need a single, static snapshot of city data. The decision depends on whether you need to perform active, filtered searches across 22 million records.

NYC 311 Service Requests: 5 Specialized Tools

These tools provide direct access to millions of NYC service requests and call center logs through natural language queries.

#	TOOL	DESCRIPTION
01	count_311_requests	This tool counts 311 service requests based on your specific filters. Use it to get a total volume of complaints before you pull individual records.
02	get_311_request	This tool retrieves a single service request using its 10-digit unique key. It automatically checks the current dataset before falling back to historic records from 2010 to 2019.
03	search_311_inquiries	This tool searches 311 call center inquiries dating back to 2014. It returns descriptions of information requests and how the call center resolved them.
04	search_311_requests	This tool searches the main 311 service request database using filters like complaint type, agency, borough, or date. It covers all requests from 2020 to the present.
05	top_311_complaints	This tool identifies the most frequent complaint types within a specific timeframe. You can adjust the number of days to look back to see recent trends.

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 top 311 complaints in the last two weeks?



The most frequent complaints over the last 14 days are: 1. Noise, 2. Street Condition, 3. Illegal Parking.

U How many open pothole complaints in Queens?



There are currently 142 open pothole complaints in Queens.

U Tell me about 311 request 70518024



Request 70518024 is a Street Condition complaint regarding a pothole in Manhattan. The status is Closed and the agency involved was the DOT.

Frequently Asked Questions

01 Do I need an API key to use this MCP?

No, this MCP works without an API key. Vinkius handles the connection to NYC Open Data for you.

02 How far back does the data go?

The current dataset covers 2020 to the present, and there is a historic dataset available for 2010 through 2019.

03 Can I search for specific boroughs?

Yes, you can filter searches by borough using uppercase names like MANHATTAN or BROOKLYN.

04 What kind of complaints can I find?

You can find data on potholes, noise, streetlights, water leaks, rodent complaints, and more.

05 Which AI clients can use this?

Any MCP-compatible client like Claude, Cursor, Windsurf, or VS Code can use this MCP.

06 Do I need an API key?

No. NYC Open Data serves all public datasets anonymously over its SODA API (data.cityofnewyork.us). This MCP defines no credentials and needs nothing configured.

07 Borough values differ across datasets — how do I know the format?

Formats are dataset-specific: 311 and shootings use UPPERCASE ("QUEENS"), food and sanitation use Title Case ("Queens"), DOT potholes use single-letter codes. Tool descriptions state the expected values, and the errors from a mismatched value are reported verbatim by the platform.

08 How far back does the 311 data go?

The main dataset covers 2020 onward (22M+ rows). Requests from 2010-2019 live in the companion historic dataset; `get_311_request` checks both, so a lookup by unique key works for either era.

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 <code>"nyc-311-service-requests": { "url": "..." }</code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → 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

NYC 311 Service Requests 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 NYC 311 Service Requests. 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	September 2026
MCP Server	NYC 311 Service Requests MCP
Server ID	01a0dfcc-039e-719a-8357-dd7197c256a8
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/en/ai-agent-connect/nyc-311-service-requests.