MCP SERVER | NO CODE | CLOUD HOSTED

New York Times MCP

Search 170+ Years of Global Journalism Data

New York Times MCP connects your AI agent directly to over 170 years of global journalism. Search archives from 1851 forward, track historical trends, pull today's top stories by section (Politics, World, Tech), and access best-seller lists and film reviews—all in one place.

A+ Quality Score 100/100

journalism | news-archive | content-search | data-retrieval | historical-data | media-api

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/new-york-times

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.

New York Times MCP

9 tools available
Cloud-hosted on Vinkius

Think about the sheer volume of information sitting across decades: breaking news reports, cultural critiques, market shifts. This MCP gives your AI agent direct access to that archive. Instead of relying on summaries or limited databases, you can query the full scope of modern journalism, cross-referencing topics and dates from 1851 right up to today.

It’s more than just reading headlines; it’s historical context in real time. You can pull top stories for a specific section, narrow your search by keywords and date ranges, or see what was most shared across social media on any given day. The Vinkius catalog makes this massive dataset available to every MCP-compatible client you use.

Whether you're writing an academic paper that needs background coverage from 1920, researching the cultural impact of a modern tech policy, or just checking out what books were trending last month, this connector handles it. It brings world-class journalism into your agent's hands.

Core Capabilities

01 — Tracking Historical Shifts

Pull articles and reports from specific years to track how a topic or event was covered over decades.

02 — Analyzing Trending Content

Determine which stories were the most viewed, shared on social media, or emailed during any 1-, 7-, or 30-day period.

03 — Searching Deep Archives

Use keywords and precise date ranges (YYYYMMDD) to find specific articles across the entire publication history.

04 — Gathering Current Headlines

Get the day's top stories instantly, filtered by major sections like World, Politics, or Technology.

05 — Accessing Cultural Data

Retrieve current and historical book best-seller lists or search for movie reviews using specific titles.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/new-york-times** — connect your AI agent in three steps.

01 Subscribe to this MCP on Vinkius and enter your New York Times Developer API Key.

02 Your AI agent uses the connection credentials to initiate a query, specifying criteria like keywords, date ranges, or sections (e.g., 'World Politics').

03 The system returns structured data containing article summaries, full text snippets, and related contextual information from the archive.

The bottom line is that your agent stops needing general internet searches and starts asking targeted questions against a verified, deep repository of journalism.

Built For

Anyone whose job involves synthesizing information across time periods. If you're tired of stitching together research from five different sources just to build context, this is for you.

Academic Researcher

Building a paper that requires comparing the media coverage of civil rights in 1960 versus 2020. You use the archive search tools to find direct comparisons.

Journalist / Content Creator

Writing an article on a current event and need background context, finding related coverage or historical parallels from previous decades for depth.

Market Analyst

Tracking how consumer sentiment regarding technology has shifted over time by cross-referencing top stories in the Tech section across different years.

What Changes When You Connect

01 Contextual depth: Use the `search_articles` tool to find articles from specific date ranges, allowing you to compare how a single event was reported decades apart.

02 Trend Spotting: Check what's popular right now. The `get_most_emailed`, `get_most_shared`, and `get_most_viewed` tools tell you exactly where the public attention is focused in any given period.

03 Broad Coverage: Need to know about books or films? You can check out best-seller lists using `get_book_lists` or find movie critiques with `get_movie_reviews`, all from one source.

04 Targeted News Retrieval: Don't waste time sifting through everything. Use `get_top_stories` to get the absolute latest headlines for a specific topic, like Politics or Tech.

05 Structural Research: Start by running `get_sections` to map out every possible content area before you even start searching for keywords.

Real-World Applications

Tracking Policy Shifts Over Time

A historian needs to understand how the coverage of climate change shifted between 1980 and 2000. They use `search_articles` with a date range, finding direct comparison pieces that show the evolution of public discourse.

Writing an Industry Deep Dive

A journalist needs to write about a tech company's rise. Instead of guessing, they use `get_top_stories` for 'Technology' and then follow up with `search_articles` to pull specific early coverage dates, building a fully sourced narrative.

Monitoring Viral Topics

A marketing team wants to know what topics generate buzz. They run `get_most_shared` for the last 7 days, immediately identifying which cultural or political subjects are currently dominating social conversation.

Patterns to Avoid

Searching everything with one keyword

✗ AVOID

Asking your agent: 'Tell me about global warming.' This is too broad and returns a mix of modern, historical, and tangential articles without focus.

✓ INSTEAD

Narrow the scope. Use `search_articles` and specify both keywords (e.g., 'Paris Agreement') AND a precise date range (e.g., 201509). This forces the agent to find highly relevant coverage.

Assuming only today’s news matters

✗ AVOID

Asking: 'What were people talking about in sports?' without limiting the time frame, resulting in a massive data dump that is unusable.

✓ INSTEAD

Specify your intent. If you want historical context, use `get_archive` to define the exact month and year you are interested in.

Mixing content types randomly

✗ AVOID

Asking: 'What was trending last week regarding books and politics?' This mixes unrelated data streams.

✓ INSTEAD

Handle them separately. First, use `get_book_lists` for the book info; then, check political coverage using `get_top_stories` or `search_articles`.

The Right Fit

Use this MCP if your goal is deep historical research or tracking cultural trends over time. You need verifiable context from a massive, established journalistic archive. Don't use it if you are looking for highly niche, non-journalistic data like real-time stock ticker prices or private internal company documents; those require specialized databases. If you just need today's breaking headlines without any historical depth, `get_top_stories` is perfect. But if the 'why' and 'how it started' are as important as the 'what happened yesterday,' this MCP gives you the necessary decades-long view.

The endless cycle of context switching

Right now, if your research requires knowing how a modern issue relates to historical coverage, you spend hours hopping between databases. You copy keywords into one search engine for 1950 data, open another terminal for best-seller lists from the same era, and finally manually check a third source just to validate the political context.

With this MCP, your agent handles all of that in one go. You ask a single question—like 'How was the Cold War covered in both 1962 and 1985?'—and you get the compiled, sourced answers directly from the archive.

Accessing Historical Context with `search_articles`

Previously, comparing coverage across different years meant manually checking archives month by month. You'd use one tool for one date range and then start the whole tedious process over again for the next time period.

Now you define a single search using `search_articles`, providing keywords and multiple dates in one command. It builds the entire timeline of coverage for you.

New York Times: 9 Available Tools

These tools allow your agent to perform specific actions, like retrieving top stories for a section or searching the entire historical archive by keywords and date.

#	TOOL	DESCRIPTION
01	`get_archive`	Retrieves every article published during a specified calendar month.
02	`get_book_lists`	Gets current or historical best-seller lists for different book categories.
03	`get_most_emailed`	Finds the articles that were most shared via email over a 1, 7, or 30 day period.
04	`get_most_shared`	Retrieves articles that saw the highest social media sharing counts in the last 1, 7, or 30 days.
05	`get_most_viewed`	Identifies the most read and viewed articles across all sections.
06	`get_movie_reviews`	Searches the archive specifically for film critiques, allowing filtering by movie title.
07	`search_articles`	Performs keyword searches across the entire database, letting you filter by date range and sort order.
08	`get_sections`	Lists every available topic or news section covered by the New York Times.
09	`get_top_stories`	Gets today's top headline stories for a specific defined section, like World or Sports.

See It in Action

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

U Show me today's top world news.

Top World Headlines: 1. Global Summit Reaches Climate Deal. 2. Elections in Europe Shift Political Landscape. 3. Tech Giants Face New Regulations in Asia.

U What is the #1 Hardcover Fiction book this week?

This Week's #1 Hardcover Fiction: 'The Midnight Library' by Matt Haig. It has been on the list for 12 weeks.

U Find movie reviews for 'The Godfather'.

Found classic NYTimes reviews: 1. Original 1972 Review by Vincent Canby — 'A magnificent film'. 2. Retrospective reviews from later decades.

Frequently Asked Questions

01 **Can I use New York Times MCP to find articles about a specific month?**

Yes, you can use the `get_archive` tool. This function retrieves all published articles within an entire calendar month for comprehensive coverage.

02 **Does New York Times MCP cover more than just news stories?**

Absolutely. Beyond top stories and archives, this MCP also includes tools for best-seller lists using `get_book_lists` and movie reviews via `get_movie_reviews`.

03 **How do I find out what was popular last year?**

You can use the trending tools. Run `get_most_shared` or `get_most_emailed`, specifying a 30-day period within the past year to pinpoint peak interest.

04 **Can I search for articles using keywords and dates in New York Times MCP?**

Yes, that's exactly what `search_articles` is for. You provide your keywords and define a precise date range (YYYYMMDD) to focus your search.

05 **What kind of sections are available in the New York Times MCP?**

You use the `get_sections` tool first. This lists all currently active topics, ensuring you know exactly which categories (like World or Sports) you can pull top stories from.

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 `"new-york-times": { "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

New York Times 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

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	New York Times MCP
Server ID	`019d75e1-b64d-70ce-8dc4-8fd753bb89e0`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE