MCP SERVER | NO CODE | CLOUD HOSTED

OpenStates MCP

Analyze US state bill status instantly.

OpenStates provides instant access to structured US state legislative data, allowing your AI agent to track bills, legislators, and committee structures across all 50 states, D.C., and Puerto Rico. Instead of navigating dozens of disparate government websites, you can query the entire network of policy action through a single connection.

A+ Quality Score 100/100

legislation | bills | us-politics | legislators | committee-data | public-policy

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

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.

OpenStates MCP

7 tools available
Cloud-hosted on Vinkius

Your agent needs deep political context, and OpenStates delivers it. It gives your AI client instant access to bills, legislators, and committee metadata across every state and major jurisdiction in the U.S., plus D.C. and Puerto Rico. You can ask your agent to check on a policy trend or verify who sponsors specific legislation without ever opening a government website. For instance, you can search for all educational proposals active in California this session, or audit the full profile of any state representative. Vinkius hosts this MCP so your agent talks to one source, no matter which compatible client you use. It's like having a dedicated political analyst constantly plugged into every state capitol building.

Core Capabilities

01 — Track and search legislation

Search for specific bills using keywords, jurisdiction, or session details to find their status and history.

02 — Audit legislator profiles

Fetch complete records, current roles, contact information, and background data on any state representative or senator.

03 — Map committee membership

List all legislative committees in a given area and retrieve the full roster of members serving on those boards.

04 — Identify supported jurisdictions

Get a list of every state and territory covered by the data to confirm coverage before starting research.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP on Vinkius and enter your OpenStates API Key.

02 Connect your preferred AI client (like Claude or Cursor) through the compatible interface.

03 Ask your agent a natural language question, like 'What bills are currently pending in Texas?'

The bottom line is that you talk to your AI client, and it handles the complex data retrieval from OpenStates automatically.

Built For

This MCP is for researchers, legal professionals, and policy teams who spend too much time manually navigating state government websites. If your job involves knowing exactly who passed what bill in which state, this is essential.

Policy Analyst

Using the MCP to monitor how specific policy topics are being debated across multiple states simultaneously.

Journalist / Investigative Researcher

Querying the history of bills and committee memberships to write detailed articles on legislative power structures.

Legal Professional (Compliance)

Verifying the current status, version number, or sponsors of state-level legislation for client work.

What Changes When You Connect

01 Stop cross-referencing state government websites. With this MCP, your agent gets all legislative data—bills, committees, and legislators—from one connection.

02 Audit any representative's background immediately. Use the `get_legislator_details` tool to pull complete profiles and current roles without manually checking state senate pages.

03 Track policy movement across multiple states. Search for education-related bills in California and New York using `search_bills`, then compare their statuses side-by-side.

04 Understand the legislative structure quickly. Use `list_jurisdictions` to see all supported states, ensuring your research scope is always correct before you begin.

05 Get deep context on specific laws. Beyond just finding a bill, use `get_bill_details` to analyze its full history and most recent actions in minutes.

Real-World Applications

Tracking state-level education reform

A policy team needs to know which states are pushing for changes regarding K-12 funding. They ask their agent to 'Search for bills related to education in the Northeast.' The agent uses `search_bills` and then pulls committee membership via `list_committees`, delivering a comparative analysis of 10 state proposals.

Verifying legal compliance

A law firm needs to know if a proposed business regulation is blocked by any current state legislation. They ask the agent to 'Check the status and sponsors for environmental bills in Arizona.' The agent uses `get_bill_details` and provides the exact version number needed for their memo.

Investigating political connections

A journalist wants to know which legislators sit on both the finance committee and the technology subcommittee. They first run `list_committees`, then use `get_committee_details` on each, giving them a definitive list of overlapping members.

Mapping political coverage

A civic tech developer needs to build a national dashboard. Instead of guessing, they first run `list_jurisdictions` to confirm which states are covered and then use the results to structure their data inputs.

Patterns to Avoid

Assuming data consistency

✗ AVOID

Trying to find a specific bill's status by only checking one state government website. You waste hours clicking through departmental silos and finding outdated information.

✓ INSTEAD

Use the OpenStates MCP. First, use `list_jurisdictions` for scope confirmation. Then, run `search_bills` to hit every relevant state database simultaneously.

Missing committee context

✗ AVOID

Finding a bill passed by a chamber and assuming it's done. You don't know which specialized committee is supposed to review it next, so you wait weeks for an update that never comes.

✓ INSTEAD

Use `list_committees` to see all relevant bodies in the state. Then use `get_committee_details` to find out who has decision-making power over that bill.

Using generic search terms

✗ AVOID

Asking your agent for 'good legislation' or 'policy ideas.' The results are useless because the data isn't filtered by specific criteria, and you get thousands of irrelevant hits.

✓ INSTEAD

Be precise. Use `search_bills` and specify keywords *and* a jurisdiction (e.g., 'public health bills in Massachusetts'). Always narrow your search.

The Right Fit

Use this MCP if you need structured, definitive political intelligence about state-level lawmaking. Specifically, if your workflow requires knowing the relationship between a bill, its sponsors, and the committees that review it, OpenStates is your tool. Don't use it if you are looking for general news coverage or opinions; it gives facts, not commentary. If you just need to verify who currently serves in state government generally, `search_legislators` works great. However, if you need a deep dive on how a specific law was passed and what happened to it afterward, you absolutely must use the combination of `get_bill_details` and `list_committees`. If your goal is purely document analysis (e.g., summarizing attached PDF policy white papers), then you're better off with a general document retrieval MCP instead.

The State Capitol Maze: Where does legislative data actually live?

Today, tracking state laws feels like trying to map a maze built by bureaucracy. To find one piece of information—say, the current status of an environmental bill in Illinois—you usually have to navigate five different government websites. You click through session indexes, cross-reference committee pages, and then try to parse PDF documents to see if it passed or failed.

With this MCP, you just ask your agent. Instead of spending hours clicking tabs and copy-pasting links between state capitol sites, you get a single, clean answer that pulls the necessary data from dozens of sources at once. You're done with manual web scraping; you get actionable intelligence.

OpenStates MCP: Get Legislative Data in One Conversation

The most time-consuming part of research is the setup: determining which committee reviewed the bill, and then finding out who was on that specific

This MCP handles all that mapping instantly. It connects disparate pieces of data—the bill's action, the committee's membership, and the

committee when it passed. You spend minutes just trying to map these connections manually.

legislator's profile—into one conversation thread. Your agent does the connecting work for you.

OpenStates: Legislative Data Access (7 Tools)

These tools allow your agent to execute specific functions like searching for bills, getting committee rosters, or retrieving detailed legislator profiles across US state records.

#	TOOL	DESCRIPTION
01	`get_bill_details`	Retrieves deep operational details and history for a specific legislative bill.
02	`get_committee_details`	Gets comprehensive information about a particular legislative committee.
03	`get_legislator_details`	Fetches the full profile and current status of an individual legislator.
04	`list_committees`	Generates a list of all available legislative committees within a specified jurisdiction.
05	`list_jurisdictions`	Provides an exhaustive list and description of all supported states and territories.
06	`search_bills`	Searches across thousands of legislative bills using keywords, jurisdiction, or session year.
07	`search_legislators`	Finds specific legislators and provides initial profile data based on names or roles.

See It in Action

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

U Search for education bills in California.

Searching OpenStates for education-related bills in California... I've identified several active proposals, including AB-150 and SB-42. I can pull the full history and current status for any of these.

U Who are the members of the New York Senate Finance Committee?

Retrieving membership for the NY Senate Finance Committee... I've found 25 legislators currently serving on this committee, including the chair and vice-chair. Shall I list their names and parties for you?

U Get the latest actions for bill TX HB-500.

Fetching details for Texas HB-500... The most recent action was 'Passed in the House' yesterday. It has now been referred to the Senate State Affairs committee. Would you like to see the full list of sponsors?

Frequently Asked Questions

01 **Can OpenStates MCP find bills from all 50 states?**

Yes, it provides intelligence across all 50 states, plus D.C. and Puerto Rico. Use `list_jurisdictions` first to confirm coverage for your specific research area.

02 **How do I find out who was on a committee?**

You can list all committees using `list_committees`, then use `get_committee_details` to retrieve the full membership roster and current roles of its members.

03 **What is the difference between searching bills and getting bill details?**

Searching bills (`search_bills`) finds a list of potential laws by keyword or state. Getting bill details (`get_bill_details`) pulls the full, deep history and most recent actions for one specific law.

04 **Does OpenStates MCP help with policy analysis?**

It provides the raw data needed for policy analysis. You can gather legislative stances on a topic across multiple states to build your argument, but you'll still write the final paper yourself.

05 **What if I need legislator info that isn’t current?**

The MCP pulls official records and profile data. If the information is not publicly filed in a state record system, the tool won't have it. Always verify the jurisdiction first.

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 `"openstates": { "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

OpenStates 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 OpenStates. 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	OpenStates MCP
Server ID	`019d8468-4041-71c1-83e8-517a1668c353`
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/openstates.