MCP SERVER | NO CODE | CLOUD HOSTED

Apache Superset MCP for AI Agents

Analyze business reports and run live SQL queries on data dashboards

Apache Superset MCP connects your AI client directly to Apache Superset. Your agent can explore BI dashboards, retrieve chart data details, and run live SQL analytics straight from chat or code. It gives you deep access into complex business reports without ever leaving your development environment.

A+ Quality Score 100/100

data-visualization | sql-lab | dashboarding | data-exploration | business-insights

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/apache-superset

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.

Apache Superset MCP

7 tools available
Cloud-hosted on Vinkius

This connector gives your conversational AI direct access to enterprise Business Intelligence tools using Apache Superset. You don't have to click through endless menus; instead, your agent indexes your entire analytical setup—from high-level operational dashboards down to specific raw data tables. Need to see how revenue was calculated? Ask the AI client for dashboard details. Want to run a custom report? Execute SQL directly against your connected databases. It handles everything from listing all available reports to pulling granular metrics and aggregating business insights on demand. If you're using Vinkius, this MCP plugs into your existing catalog, making Superset analysis one of many powerful tools your agent can access.

Core Capabilities

01 — List Available Reports

Shows a comprehensive list of all dashboards and charts currently built in Apache Superset.

02 — Get Dashboard Details

Retrieves the specific configuration, metrics, and underlying data for any given dashboard ID.

03 — List Data Sources

Identifies all connected database connections (like Postgres or MySQL) used by Superset.

04 — Run Custom SQL Queries

Executes specific, raw SQL statements against a chosen database connection ID to generate custom reports.

One Click on Vinkius — From Prompt to Execution

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

01 Append the Apache Superset MCP module into your agent's operational integrations panel.

02 Configure your AI client by providing the active `SUPERSET_BASE_URL` and a validated `SUPERSET_ACCESS_TOKEN`. This connection authenticates your session with Superset.

03 Ask your agent to perform an analytic task, such as: "List all dashboards and then run a query for Q3 sales figures."

The bottom line is that once configured, your AI client treats the entire BI portal like another API endpoint, allowing direct data interaction via conversation.

Built For

This MCP is built for power users who live in data. If you're a Data Analyst spending hours clicking through dashboards to validate numbers, or a Product Manager who needs to check metric shifts without involving the BI team, this tool saves time and clicks.

Data Analyst

Uses the MCP to test complex SQL queries and audit dashboard definitions in real-time, validating data pipelines before they go live.

Product Manager

Checks key performance indicators (KPIs) by asking for specific dashboard details or running ad-hoc reports on user behavior metrics via the chat interface.

Data Engineering Lead

Probes backend storage clusters and checks dataset definitions to isolate data anomalies or test new semantic layers without needing full GUI access.

What Changes When You Connect

01 You can validate complex metrics instantly. Use `get_chart_details` to see exactly how a number was calculated, instead of just trusting the dashboard display.

02 Audit your entire BI infrastructure without logging into Superset's UI. Running `list_dashboards` gives you an immediate inventory of what reports exist.

03 Skip manual data extraction steps. With `execute_sql_query`, your agent runs raw SQL against production databases and returns the resulting table directly to chat.

04 `list_databases` lets you audit all connected sources at a glance, which is critical for data engineers tracking connectivity changes.

05 Understand dashboard relationships instantly. Calling `get_dashboard_details` shows you the parent-child relationship between charts without needing to click around.

Real-World Applications

Validating a KPI in a live report

A Product Manager needs to confirm if 'Monthly Active Users' on the main dashboard is using the right definition. They ask their agent, and it uses `get_dashboard_details` followed by `list_charts` to pinpoint the exact underlying metric logic for validation.

Deep-diving into quarterly revenue gaps

A Data Analyst suspects a data source issue. They use `list_databases` first, then run `execute_sql_query` on a specific connection to pull raw transaction logs and find the discrepancy manually.

Inventorying all reporting capabilities

A new team member needs to know what reports exist. They prompt their agent to use `list_dashboards` and then `list_datasets` to get a full, categorized map of the entire BI portal.

Checking for deprecated metrics

A Data Engineer suspects an old dataset is unused. They prompt their agent to use `list_datasets` and check which datasets are referenced by existing charts using `get_chart_details` to confirm if it's safe to retire.

Patterns to Avoid

Guessing the data structure

✗ AVOID

Trying to write a complex SQL query without knowing which tables or connections are available, resulting in a connection error.

✓ INSTEAD

First, always run `list_databases` to identify active source connections. Then use `list_datasets` to confirm table names before writing your raw extraction with `execute_sql_query`.

Overlooking dashboard dependencies

✗ AVOID

Attempting to analyze a chart's data without knowing which metrics feed it, leading to incomplete analysis.

✓ INSTEAD

Before analyzing any chart, use `get_dashboard_details` and then `get_chart_details` to trace back the full dependency chain for accurate context.

Forgetting available reports

✗ AVOID

Wasting time manually checking old internal documentation instead of seeing what's actually available in the BI tool.

✓ INSTEAD

Start by calling `list_dashboards` to get a definitive, up-to-date inventory of all existing operational reporting surfaces.

The Right Fit

Use this MCP if your primary bottleneck is turning dashboard data into actionable reports. If you need to run ad-hoc SQL queries or audit the underlying metrics definition for dashboards—this is what you want. Don't use it if you simply need to *visualize* data; the AI client handles that. Furthermore, don't use this MCP if your goal is purely ETL (Extract, Transform, Load) pipeline management; those are dedicated orchestration tools. If you only need a basic list of reports and never need to run custom SQL or check metrics details, simpler reporting APIs might suffice, but this one gives you the deep context needed by professionals.

Apache Superset MCP for AI Agents: Solving Dashboard Metric Auditing

Right now, when a stakeholder asks 'Where did that number come from?', you're stuck. You have to manually pull up the dashboard in your browser, click on the chart, find the metric definition panel, and copy the underlying SQL or formula—a tedious cycle of clicks and context switching.

With this MCP, asking for metrics is a simple chat command. Your agent calls `get_chart_details` and hands you the precise calculation logic immediately. You get the raw truth about any number on any dashboard without ever leaving your terminal.

Apache Superset MCP for AI Agents: Streamlining Data Source Discovery

Manually understanding a data stack involves logging into multiple systems just to check connectivity. You have to jump between the BI tool and the source database connection panel, wasting time verifying if 'SalesDB' is even live or what type of credentials it uses.

Now, you ask your agent to list connections using `list_databases`. It pulls that entire metadata sheet for you. You instantly know every active data source, its ID, and its status in one clean output.

7 Tools in Apache Superset for Data Visualization Analytics

These tools allow your agent to list reports, check dashboard details, inspect datasets, and execute complex raw SQL queries against your data sources.

#	TOOL	DESCRIPTION
01	`execute_sql_query`	Runs a specific SQL query against a selected database connection ID, returning the raw result set.
02	`get_chart_details`	Pulls all metadata about how a single chart (or slice) is built, including its metrics and filters.
03	`get_dashboard_details`	Retrieves the full structure of a dashboard, showing which charts are included and their overall purpose.
04	`list_charts`	Provides an inventory of every chart or visualization available within your Superset environment.
05	`list_dashboards`	Lists all user-facing dashboards, giving you a quick overview of the reporting surface area.
06	`list_databases`	Displays a list of all active data source connections that Superset uses for its reports.
07	`list_datasets`	Inventories every unique dataset available, helping you map out the entire analytical structure.

See It in Action

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

U What's the revenue performance breakdown by product category this quarter?

Q3 Revenue Breakdown

CATEGORY	TOTAL SALES	% OF REVENUE
Electronics	$1.2M	45%
Apparel	$950K	35%
Software	$410K	15%
Home Goods	$620K	25%

Note: Data pulled from the 'Main Sales DB' connection.

U Show me the details for the 'User Funnel' dashboard and tell me which charts it uses.

Dashboard: User Funnel

This dashboard tracks user journeys across four stages. It includes:

- Chart 1: Registration Count (Metric: `new_users`)
- Chart 2: First Login Rate (Source: `user_activity`)
- Chart 3: Conversion Path Map (Type: Funnel Diagram)

All metrics are sourced from the 'Core Metrics' dataset.

U List all datasets and tell me which ones connect to Postgres.

I found 12 available datasets. Here is a breakdown of their connections:

- `sales_transactions`: PostgreSQL (Primary)
- `user_activity`: Snowflake
- `marketing_campaigns`: PostgreSQL (Secondary)
- ... and so on.

The two key PostgreSQL sources are `sales_transactions` and `marketing_campaigns`.

Frequently Asked Questions

01 **How does the Apache Superset MCP help me audit dashboard metrics?**

The MCP lets your agent retrieve granular details on any chart, showing you the exact metric and underlying data source used. You can trace a number back to its origin without needing manual UI navigation.

02 **Can I run custom reports using Apache Superset MCP for AI Agents?**

Yes. By executing raw SQL queries through the agent, you bypass the dashboard's built-in filters and write exactly what data you need directly against the connected databases.

03 **What if I need to know which dashboards are available right now?**

You can use the MCP to list all existing dashboards. This gives you an immediate, comprehensive inventory of every reporting surface built in your Superset instance.

04 **Is the Apache Superset MCP for AI Agents good for data engineering tasks?**

Absolutely. You can use it to audit database connections (`list_databases`) and inspect dataset metadata, helping you spot anomalies or confirm connectivity status quickly.

05 **Does this MCP support multiple data sources?**

Yes, the system manages and lists all connected databases. You can select any active connection ID to run your custom SQL query against a different data source.

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 `"apache-superset": { "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

Apache Superset 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 Apache Superset. 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	Apache Superset MCP
Server ID	`019d760f-2ed4-72f6-8f02-129162b57964`
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/apache-superset.