MCP SERVER | NO CODE | CLOUD HOSTED

Alpaca Trading MCP for AI Agents

Execute stocks and crypto trades with real-time market data access

Alpaca Trading gives your AI agent direct, programmatic access to real-time stock, crypto, and brokerage data. It lets you execute complex trades—placing market, limit, or stop orders—and manage account details without leaving your chat window. Use it to pull historical bars for backtesting, monitor live quotes, or check every order's status instantly.

F Quality Score 3.6/100

algorithmic-trading | stock-market | crypto-trading | brokerage-api | market-data | financial-automation

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

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.

Alpaca Trading MCP

14 tools available
Cloud-hosted on Vinkius

Connect Alpaca Trading to your AI agent and automate everything from trade execution to deep market analysis. This MCP lets you treat your AI client like a sophisticated trading terminal capable of interacting with real brokerage accounts. Instead of manually logging into a dashboard, asking questions or running scripts in natural language sends commands directly to the exchange.

Need to know if shorting is allowed? Use one command. Need to place an order for 10 shares when the price hits $50? That's another. The system handles the complexity behind the scenes. When you subscribe, you connect your specific Alpaca API keys and instantly get access to this entire suite of financial tools through Vinkius' catalog.

This means whether you're a retail investor checking portfolio status or a developer integrating live market data into an application, your agent acts as the central command post. You can retrieve historical quotes, check current asset availability, and manage account configurations—all conversational actions that used to require multiple logins and complex API calls.

Core Capabilities

01 — Execute and Manage Trades

Place new trading orders (market, limit, or stop) and clear all outstanding open orders with a single prompt.

02 — Retrieve Real-Time Market Data

Get the latest quotes, current trades, and historical bars for both stocks and cryptocurrencies.

03 — Monitor Order History

Query all your past and active orders using filters like status or symbols to track performance.

04 — Manage Account Configuration

Check your current brokerage account details and update settings, such as shorting permissions or fractional trading options.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and enter your required Alpaca API Key ID and Secret Key.

02 Your AI agent uses natural language instructions (e.g., 'Buy 5 shares of AAPL at market price') to determine the necessary action.

03 The system executes the command, returning confirmation details or requested market data directly into your chat.

The bottom line is that you're running complex financial operations using only conversation with your AI client.

Built For

Anyone who spends time juggling multiple screens—a quant trader needing historical data for backtesting, a developer building trading applications, or an experienced retail investor executing trades on the fly. It's for people whose workflow is currently broken by manual copy-pasting and API calls.

Quantitative Researcher

Requires pulling historical bars using `get_stocks_bars` to build backtesting models and analyze asset performance across different timeframes.

DevOps Developer

Integrates live financial data feeds into custom applications or scripts by calling tools like `get_latest_stocks_quotes` for real-time reporting.

Active Retail Investor

Checks account status and places immediate, conversational trades using simple prompts without logging out of their primary chat interface.

What Changes When You Connect

01 You can place complex orders using `create_order` or cancel everything instantly with `delete_all_orders`, letting your agent handle the transaction logic.

02 Track every trade's status, from open to closed, by querying your order history directly with `get_orders`. You always know what's active.

03 Analyze market trends quickly. Use `get_stocks_bars` or `get_crypto_bars` to pull historical data for immediate backtesting analysis.

04 Get instant pricing updates on stocks via `get_latest_stocks_quotes`, so your agent can quote current values without needing a separate API call.

05 Manage risk and compliance by checking and updating account rules using `update_account_configs` before running automated strategies.

Real-World Applications

Backtesting an old strategy

A quant researcher needs to test a volatile pattern from six months ago. Instead of writing complex code, they ask their agent for the historical data using `get_stocks_bars`, and the agent pulls the exact time series needed immediately.

Reacting to breaking news

When a company announces unexpected earnings, the investor prompts their agent: 'What's the current quote for AAPL?' The agent uses `get_latest_stocks_quotes` and reports the live bid/ask spread instantly.

Executing a complex multi-leg trade

The user dictates, 'I want to buy 10 shares of NVDA if it drops below $90.' The agent interprets this as a limit order and executes the setup using `create_order` without manual intervention.

Checking all open positions before quitting for the day

The user simply asks, 'What orders are currently pending?' The agent runs `get_orders`, providing a clean summary of every active trade and its current status.

Patterns to Avoid

Trying to predict price swings

✗ AVOID

Thinking that simply listing available assets via `get_assets` will tell you which stocks are going to rise next week. This only gives a list, not predictive insight.

✓ INSTEAD

To analyze potential movements, use the historical data tools like `get_stocks_bars`. You need concrete time-series information to spot patterns that might suggest future movement.

Manually tracking every order

✗ AVOID

Opening the brokerage portal and clicking through dozens of tabs just to see if a limit buy or sell order was filled yesterday. This is tedious and error-prone.

✓ INSTEAD

Use `get_orders`. You can filter by status, symbol, or time frame with one prompt, getting a clean list without navigating multiple web pages.

Forgetting account constraints

✗ AVOID

Trying to run an automated strategy that requires shorting stock X when the user hasn't updated their permissions. The system will fail because of missing configuration.

✓ INSTEAD

Always check your settings first. Use `update_account_configs` or `get_broker_account` to ensure you have the necessary trading permissions enabled before executing major strategies.

The Right Fit

Use this MCP if your workflow involves constant interaction with live market data, requiring immediate execution of trades (stocks or crypto), and needs automated order tracking. It’s perfect for anyone building quantitative models that need real-time endpoints.

However, don't use it if you only need static financial reporting—for instance, generating a PDF quarterly report based on settled accounts. For that, an accounting software integration is better. Also, if your primary need is just viewing basic portfolio summaries without executing trades, the dedicated brokerage dashboard might suffice. But when automation and real-time action are key, this MCP provides the necessary deep access.

Alpaca Trading: Automating Stock Order Execution with AI Agents

Today, placing even a simple trade requires multiple steps. You log into the brokerage site, search for the ticker, select order type (market or limit), input shares, and hit 'submit.' If you're building an application, that means writing complex API calls, handling authentication tokens, and managing error states—all before you even get to the data.

With this MCP, the process collapses. You just tell your agent what you want to do: 'Sell 50 shares of MSFT at market price.' The system handles the order creation, validation, and submission. What you get is instant confirmation that the action was taken.

Alpaca Trading: Retrieving Crypto Market Intelligence using AI Agents

Getting market intelligence used to mean toggling between different data feeds. You'd check a quote service for the current price, then switch to a charting platform to pull historical bars, and finally open a separate sheet to calculate performance metrics—a time-consuming mess of copy/pasting.

Now, your agent pulls everything together. Ask it for the latest quotes on BTC/USD and then request the last 90 days of historical data. It delivers both sets of information immediately, giving you clean, actionable intelligence in one go.

Alpaca Trading: 14 Tools for Financial Market Automation

Use these tools to create accounts, place orders, get the latest quotes, retrieve historical bars, and manage all aspects of stock and crypto trading through your AI agent.

#	TOOL	DESCRIPTION
01	create_broker_account	Initiates the process to set up a brand new brokerage account with Alpaca Markets.
02	create_order	Places a specific trading order for stocks or crypto assets, defining the type and quantity.
03	delete_all_orders	Cancels every single open order associated with your current account ID.
04	get_assets	Lists all assets that are currently available and tradable on the Alpaca exchange platform.
05	get_broker_account	Retrieves specific details about your brokerage account using its unique identifier.
06	get_crypto_bars	Fetches historical price bars and data points specifically for cryptocurrency pairs.
07	get_latest_stocks_quotes	Retrieves the most current bid and ask prices for specified stock tickers.
08	get_latest_stocks_trades	Gets details on the most recent executed trades that have occurred in specific stocks.
09	get_orders	Queries and lists all of your current and past orders, allowing for advanced filtering by status or time frame.
10	get_stocks_bars	Retrieves historical price bars (OHLCV data) for specified stock tickers across different time periods.
11	get_stocks_quotes	Gets historical quote records for stocks, useful for analyzing pricing patterns over time.
12	get_stocks_trades	Retrieves a log of past executed trades (price and volume) for specific stock tickers.
13	issue_token	Generates secure OAuth2 tokens necessary for automated machine-to-machine authentication.

#	TOOL	DESCRIPTION
14	update_account_configs	Modifies various account settings, such as enabling or disabling fractional trading permissions.

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 current latest quotes for AAPL and ETH?

Market Quotes (Real-Time)

TICKER	ASSET	BID PRICE	ASK PRICE
AAPL	Stock	$175.20	$175.45
ETH/USD	Crypto	$3,450.10	$3,452.00

These quotes are live and subject to immediate market change.

U I want to buy 20 shares of TSLA when the price hits $180.

✅ **Order Placed Successfully**

We've set up a limit order for 20 shares of TSLA.

- **Type:** Limit Buy
- **Target Price:** $180.00
- **Status:** Pending
- **Order ID:** 550e8400-e29b

You'll get an alert when the price hits this level.

U Show me all my closed orders from the last week.

Closed Order Summary (Last 7 Days)

You had 3 transactions:

- **MSFT:** Bought 15 shares. Filled at $420.10.
- **TSLA:** Sold 10 shares. Filled at $225.99.
- **NVDA (Limit):** Bought 5 shares. Filled exactly at your set limit price of $880.00.

Frequently Asked Questions

01 **How do I use the Alpaca Trading MCP to place a trade?**

You simply tell your AI agent what you want to buy or sell, and how many shares. You don't need to worry about order types or syntax; the agent translates your natural language command into an executable trading instruction.

02 **Can this MCP help me analyze crypto data?**

Yes, it handles crypto just as well as stocks. You can retrieve historical price bars and get the latest quotes for major cryptocurrencies like Bitcoin or Ethereum, making deep market analysis possible.

03 **Is Alpaca Trading MCP safe to use for live trades?**

The MCP connects directly to your brokerage account using secure API keys. It ensures that every order you place is tracked and executed according to the rules of your connected Alpaca account.

04 **What if I need historical data for backtesting? Does Alpaca Trading support it?**

Absolutely. You can pull specific historical records, such as daily or hourly price bars, for any stock or crypto pair. This allows you to test trading strategies against real past market conditions.

05 **Does the MCP help me manage my account settings?**

Yes. If you need to change permissions, like enabling fractional shares or adjusting shorting limits, your agent can interact with your account configuration using simple commands.

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 `"alpaca-trading": { "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

Alpaca Trading 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 Alpaca Trading. 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	Alpaca Trading MCP
Server ID	`019e3861-b8c5-72f1-84b6-bf6cab07f797`
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/alpaca-trading.