

MCP SERVER

NO CODE

CLOUD HOSTED

# Order Time MCP

Audit inventory and track complex order flows.

Order Time MCP connects your AI agent directly to professional inventory, customer, and fulfillment data. You can list full product catalogs, check detailed item specs, find specific customers, and track both sales and purchase orders with natural language commands.

**A+** Quality Score 100/100

inventory-tracking

order-fulfillment

purchase-orders

sales-tracking

warehouse-management



# 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](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Order Time MCP

9 tools available

Cloud-hosted on Vinkius

Managing physical goods means juggling three things: what you sell, who you sell it to, and where the inventory sits right now. This MCP gives your AI agent programmatic access to all of that data via Order Time's API. Instead of opening ten browser tabs—one for sales, one for customers, and a dozen for stock counts—you tell your client exactly what you need, and it retrieves it instantly.

Need to know if Customer X paid on time? Your agent pulls their profile using the `get_customer` tool. Need to verify that 50 units of Part Y are actually in the warehouse? Use `list_items`. Then, trace back whether those items were tied up in a recent sales order or waiting for a purchase delivery. This unified view means your AI client can do complex cross-checks—for instance, comparing current stock levels against pending orders—without you writing a single SQL query. By connecting this MCP through Vinkius, you give your agents the full operational picture of your business, letting them handle everything from simple inventory queries to complete fulfillment audits.

---

## Core Capabilities

### 01 — View and audit product stock

You can list every item in your catalog or get detailed specs for a single part number.

### 03 — Monitor sales and purchase movements

Check the real-time status of both completed and pending sales orders, as well as incoming purchase deliveries.

### 02 — Track customer records

Retrieve comprehensive profiles, including historical details, for any known client.

### 04 — Pull specialized metadata

Access custom field definitions attached to items or customers for niche business data.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/order-time](https://vinkius.com/mcp/order-time) — connect your AI agent in three steps.

- 01 Subscribe to this MCP on Vinkius and enter your Order Time API Key, User Email, and Password.
- 02 Connect your preferred AI client (Claude, Cursor, etc.) to the new connection endpoint.
- 03 Ask your agent a question like, 'What is the current stock level for part number 90210 and which customer was it meant for?' The MCP handles the data retrieval.

The bottom line is, you hand off complex data lookup tasks to your AI client instead of manual dashboard navigation.

---

## Built For

This connector is essential for Operations Managers, Supply Chain Analysts, and E-commerce fulfillment teams. If you spend too much time manually cross-referencing stock sheets with sales reports across multiple tabs, this MCP saves your sanity.

### Operations Manager

Needs to quickly check if incoming purchase orders contain enough material to fulfill a high-priority sales order.

### Warehouse Supervisor

Must verify current inventory levels against the expected count in real time, logging discrepancies immediately.

### Sales Analyst

Needs to quote a client and instantly validate if the item is stocked and available for shipment today.

---

## What Changes When You Connect

- 01 You immediately know if an item is stocked. Instead of calling the warehouse, asking your agent to run `get_item` gives you instant stock levels for any part number.

- 
- 02** Cross-reference sales against purchases easily. Your agent can take a list from `list_sales_orders` and check that same data point against pending incoming shipments using `list_purchase_orders` .
- 
- 03** Stop chasing records. Need to know everything about Customer Smith? Running `get_customer` pulls their full profile, replacing multiple database lookups into one simple prompt.
- 
- 04** Improve auditing speed by getting a complete item list in bulk. The `list_items` tool provides immediate visibility across your entire product catalog, helping you spot low stock before it's too late.
- 
- 05** Understand the history behind every transaction. By checking both `get_sales_order` and `get_purchase_order` , you build a clear paper trail showing why an item is in or out of stock.
- 

---

## Real-World Applications

### **A sales rep needs to confirm product availability for a large quote.**

The agent runs `get_item` on the specific SKUs listed. It confirms that 50 units are available and pulls up customer contact details using `get_customer`, allowing the rep to send an accurate, immediate quote.

### **A customer calls asking about a past order's status.**

The agent takes the provided Order ID and runs `get_sales_order`. It returns all items purchased, the billing address, and the current fulfillment stage without needing human intervention.

### **An operations team needs to audit a shipment delay.**

The analyst uses `list_purchase_orders` to find the expected delivery date. Then they run `get_purchase_order` for that specific PO number, identifying which internal department was responsible for the final receiving check.

### **A store manager needs a complete inventory count for annual tax purposes.**

The agent executes `list_items` to get every SKU name and its exact quantity on hand. It can then append this data with any custom field definitions using `list_custom_fields` for enhanced reporting.

---

# Patterns to Avoid

---

## Asking for everything at once

### X AVOID

Prompting the agent: 'Give me all sales, all purchases, and every single item detail.' The response will be a massive data dump that's impossible to read or act upon.

### ✓ INSTEAD

Focus your query. If you need stock details, use ``list_items`` first. If you need order status, start with ``list_sales_orders``, then narrow it down by ID using ``get_sales_order``.

---

## Assuming the data format

### X AVOID

Telling the agent to 'find all items related to client ABC.' The agent won't know if you mean a sales order or just general inventory, leading to errors.

### ✓ INSTEAD

Be explicit. Specify the tools in your prompt. Ask: 'List all sales orders for customer XYZ using ``get_sales_order`` and then check that item against current stock via ``list_items``.'

---

## Confusing order types

### X AVOID

Asking the agent to 'check my next shipment' without specifying if it's a sale or an incoming supply delivery. The AI will guess, which is usually wrong.

### ✓ INSTEAD

Always specify the document type: Use ``list_purchase_orders`` for deliveries coming in, and use ``list_sales_orders`` for goods going out.

---

## The Right Fit

Use this MCP if your business relies heavily on managing physical inventory or processing complex transactions involving multiple parties (customer sales, vendor purchases). You need to know the definitive status of a product—is it stocked, is it ordered, or has it shipped? If your primary need is purely administrative, like updating employee payroll records or handling HR data, this MCP won't help. For those kinds of tasks, you need an HCM (Human Capital Management) type tool instead. However, if you need to know *why* the inventory count is wrong—for instance, comparing a pending order ( `get_purchase_order` ) against what was actually sold ( `get_sales_order` )—this MCP provides the specific tools needed to make that comparison.

---

## The manual work of reconciling stock and sales records is brutal.

Right now, tracking inventory means jumping between a CRM tab for customer billing, an accounting system for purchase orders, and a dedicated warehouse dashboard just to see if the item exists. You end up copy-pasting SKU numbers, cross-referencing dates in Excel, and spending hours reconciling discrepancies that should take minutes.

With this MCP connected via Vinkius, your agent handles the data plumbing. Instead of manual lookups, you simply ask: 'Show me all sales orders for customers who haven't ordered anything in 90 days.' You get a clean list, instantly generated from `list_sales_orders` and customer records.

---

## Order Time MCP delivers unified data visibility.

The biggest time sink goes away: the manual process of tracking down a single product's entire lifecycle. You no longer have to check if the item was purchased, received, sold, or is currently sitting in stock; all that status information converges into one query.

The data flow changes completely. What used to require three different logins and half an hour of focused effort now takes a single question and an instant response from your agent.

---

# Order Time MCP with 9 Tools

These tools allow your AI client to access specific functions—from listing customers to checking item inventory—to perform deep data lookups on demand.

#	TOOL	DESCRIPTION
01	<code>get_customer</code>	Retrieves all detailed information about a specific customer by their ID or name.
02	<code>get_item</code>	Fetches the full specifications, current stock count, and details for one item number.
03	<code>get_purchase_order</code>	Gets all data associated with a specific purchase order ID, including line items and expected delivery dates.
04	<code>get_sales_order</code>	Retrieves the complete details for one sales order, showing who it was billed to and what products were included.
05	<code>list_custom_fields</code>	Shows a list of specialized data fields you can retrieve from your records, like custom tracking IDs.
06	<code>list_customers</code>	Returns a browsable list of all customers in your system, perfect for finding contact info quickly.
07	<code>list_items</code>	Provides a comprehensive catalog view, listing thousands of items and their current inventory counts.
08	<code>list_purchase_orders</code>	Lists multiple purchase orders, allowing you to see which incoming deliveries are scheduled for the week.
09	<code>list_sales_orders</code>	Returns a list of recent sales orders, giving an overview of all transactions that happened today or this month.

---

## See It in Action

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

### **U** List my inventory items.



Fetching items... I've found 150 items in your catalog. Here are the most recent ones...

---

## Frequently Asked Questions

### **01** How do I check the current stock levels using Order Time MCP?

To see all products, use ``list_items``. If you need details on one specific part number, run ``get_item`` and specify the item ID.

### **02** Can I track both incoming and outgoing orders with Order Time MCP?

Yes. Use ``list_sales_orders`` for items leaving your facility. Use ``list_purchase_orders`` to see everything scheduled to arrive at your warehouse.

### **03** What if I need historical data on a customer? Does Order Time MCP support that?

The agent pulls all relevant history using ``get_customer``. This includes past order details and contact records, letting you understand their full relationship with your business.

### **04** Does the Order Time MCP let me see custom product data?







Yes. The ``list_custom_fields`` tool lets you retrieve specialized metadata that isn't in standard fields, perfect for compliance or tracking unique identifiers.

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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
 <b>Claude AI</b>	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
 <b>Cursor</b>	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
 <b>VS Code</b>	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add <code>"order-time": { "url": "..."</code>
 <b>Windsurf</b>	MCP Settings → <code>mcp_settings.json</code> → Add endpoint URL
 <b>ChatGPT</b>	Settings → Tools & plugins → Add MCP server → Paste endpoint
 <b>Gemini</b>	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

# Order Time 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](https://cloud.vinkius.com) →

[vinkius.com](https://vinkius.com) · [support@vinkius.com](mailto: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 Order Time. 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	Order Time MCP
Server ID	019d75ec-7430-70ae-9181-e24bbb98e062
Platform	Vinkius Cloud for AI Agents
Endpoint	<a href="https://edge.vinkius.com/{token}/mcp">https://edge.vinkius.com/{token}/mcp</a>

### 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/order-time](https://vinkius.com/mcp/order-time).