MCP SERVER | NO CODE | CLOUD HOSTED

BoxHero MCP for AI Agents

Track Stock Levels and Warehouse Inventory Management

BoxHero connects your inventory management to any AI agent. List items, track every stock transaction—whether it's a shipment in or an item moved location—and keep precise records of storage locations using natural conversation.

F Quality Score 3.11/100

stock-tracking | barcode-scanning | inventory-control | transaction-history | warehouse-management | small-business-tools

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

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.

BoxHero MCP

10 tools available
Cloud-hosted on Vinkius

Running a physical store or warehouse means dealing with constant movement: things coming in, getting put away, and eventually shipping out. BoxHero lets your AI agent manage that complexity without you ever touching a dashboard. Instead of digging through spreadsheets to see if 'Keyboard' stock levels are accurate, you just ask your agent. It immediately knows the current count, the full transaction history, and where those items are physically stored in your facility.

It handles everything from creating new item records to logging complex movements using structured data. The entire process is orchestrated through natural conversation. You connect BoxHero via Vinkius, giving your AI agent instant access to all your inventory records—item specifications, location details, and every stock movement that has ever happened. It turns what used to be a massive manual audit into a simple query.

Core Capabilities

01 — Inventory Item Creation and Modification

Registers a brand new item that has not yet been tracked in your inventory system, or updates existing product details.

02 — Stock Movement Logging (In/Out/Move)

Logs every physical movement of units, recording if they were added (stock-in), removed (stock-out), or transferred internally.

03 — Catalog Detail Retrieval

Pulls all specific data points for one item by knowing its unique identifier, and retrieving custom attributes across the catalog.

04 — Warehouse Mapping and Listing

Provides an overview of all designated storage locations or generates a full list of every managed inventory item.

05 — Historical Data Reporting

Retrieves comprehensive lists of past transactions, showing exactly when and why stock levels changed over time.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to the BoxHero MCP on Vinkius.

02 Enter your unique BoxHero API Token into your AI client's settings.

03 Tell your agent what you need—for example, 'Show me all inventory that needs a count audit,' and it executes the necessary operations.

The bottom line is that once connected, your AI agent handles the connection details and API calls; you just talk to it naturally.

Built For

This MCP is for anyone running a physical inventory or fulfillment operation. If your job involves knowing exactly how many units of a product are in stock, where they are located, and when they moved there, this tool saves you hours of manual data reconciliation.

Warehouse Manager

Using the MCP to audit stock levels across multiple locations or initiating bulk adjustments for cycle counting.

Operations Planner

Checking historical transaction logs to predict supply chain bottlenecks or optimizing item placement in storage areas.

Small Business Owner/Retail Manager

Quickly listing all available items and retrieving custom attributes for merchandising reports without logging into the main dashboard.

What Changes When You Connect

01 Eliminate manual stock audits. Use the `list_transactions` tool to pull a complete history of every item move, giving you instant audit readiness.

02 Never lose track of where things are. The MCP lets you list all storage locations via `list_locations`, ensuring your AI agent knows which warehouse floor to check.

03 Speed up onboarding new products. Use `create_item` and `list_attributes` together, letting the AI build out item records with custom metadata in seconds.

04 Accurate counts are always available. The system allows you to use `get_item` alongside transaction logging (`create_transaction`) for real-time stock verification.

05 Streamline updates. Instead of manually changing data, use `update_item` through natural language conversation to modify item descriptions or names.

Real-World Applications

Need a full inventory count before shipping?

The warehouse team asks the agent: 'I need a total count of all monitors.' The agent runs `list_items` to find the SKU and then uses transaction history functions to calculate the precise, real-time stock level.

A shipment arrived late and needs logging.

The receiving clerk asks: 'Log 50 units of Keyboard that just arrived.' The agent executes `create_transaction` (Stock In), ensuring the item count is accurate immediately upon arrival.

We need to know if an item was ever moved.

The manager asks: 'Show me all movements for SKU X.' The agent uses `list_transactions` and filters by location, providing a clear audit trail of the product's history.

We are reorganizing the warehouse layout.

Instead of physically mapping every area, the user asks the agent to list all storage locations. This provides an immediate overview needed for planning and zoning changes.

Patterns to Avoid

Updating inventory without logging movement

✗ AVOID

Trying to use a basic update feature to change stock count. If you just manually set the number, you lose the audit trail—you don't know *why* the count changed.

✓ INSTEAD

Always log changes using `create_transaction`. Use this tool to record the movement (stock-out or stock-in) first; then, if necessary, use `update_item` for non-count details.

Listing items only by name

✗ AVOID

Asking the AI agent 'What monitors do we have?' without specifying a location. The response might be inaccurate because it doesn't know if you mean the main warehouse or the store room.

✓ INSTEAD

Always include location context in your query, or first run `list_locations` to confirm the scope of your audit.

Forgetting item attributes

✗ AVOID

Creating a new product record but forgetting key details like color variation or weight. This makes it impossible for agents to search accurately later.

✓ INSTEAD

Always use `list_attributes` first, then ensure the necessary custom fields are populated when you run `create_item`.

The Right Fit

Use this MCP if your business relies on a precise audit trail. If knowing *when* and *how* an item's count changed is as important as the current count itself, BoxHero handles that history for you using `list_transactions`. However, don't use it just because you need to list things; if all you need is a static catalog lookup without tracking physical movement, a simpler database MCP might suffice. Also, remember that this tool focuses on core inventory operations—it doesn't handle financial accounting or invoicing, so those processes require separate integration.

BoxHero Inventory Management: Solving Stock Count Discrepancies

Today, checking stock levels means jumping between the online catalog and physical manifests. You pull up a spreadsheet, cross-reference it with warehouse reports, and manually compare counts for every item—a process that is slow, prone to human error, and requires hours of dedicated auditing time.

With BoxHero connected via your AI agent, you simply ask: 'What's the current stock count for all wireless mice?' The MCP runs through its tools, checking real-time data and showing you a definitive answer instantly. You get immediate accuracy without touching any dashboards.

BoxHero Stock Tracking: Managing Warehouse Location Changes

Manual warehouse management involves paper checklists, sticky notes, and physical searches just to know which aisle a product moved to. If an item is misplaced or the location record is wrong, it stops moving until someone physically finds it.

This MCP eliminates that guesswork. You can ask your agent to list all storage locations (`list_locations`) or even run `create_transaction` for a move from Aisle 5 to Shelf B. The system updates the item's location record instantly, giving you perfect inventory visibility across the entire facility.

BoxHero 10 Tools for Inventory & Transaction Operations

Use these tools to create item records, log every physical movement, list current stock levels, and map out your entire storage facility with AI agents.

#	TOOL	DESCRIPTION
01	`create_item`	Registers a brand new item that has not yet been tracked in your inventory system.
02	`create_transaction`	Logs a stock movement, recording if units were added (in), removed (out), or moved internally.
03	`delete_item`	Permanently removes an item record from your inventory catalog.
04	`get_account_info`	Retrieves basic details about the user who authenticated access to the system.
05	`get_item`	Pulls all specific data points for one item by knowing its unique identifier.
06	`list_attributes`	Retrieves a list of custom fields or metadata you've defined across your product catalog.
07	`list_items`	Generates a full list of every item currently managed within the inventory system.
08	`list_locations`	Provides an overview of all designated storage areas and physical warehouses connected to your account.
09	`list_transactions`	Lists past stock movements, showing records of items coming in or going out over time.
10	`update_item`	Modifies existing product details, such as renaming an item or adjusting its description.

See It in Action

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

U I need an audit report on all 'Keyboard' units: how many do we have right now and what were their last five movements?

Inventory Audit Report

- **Item:** Keyboard (SKU KBD-001)
- **Current Stock Level:** 80 units
- **Last Five Transactions:**

* ✅ **2024-06-13:** +50 units (Stock In - Supplier Alpha)
* ❌ **2024-06-12:** -10 units (Stock Out - Order #998)
* 🔄 **2024-06-11:** +5 units (Move - From Receiving to Aisle 3)
* ✅ **2024-06-10:** +75 units (Stock In - Shipment Beta)
* ❌ **2024-06-09:** -15 units (Stock Out - Order #997)

U List all active items that have custom attributes, like 'warranty type' or 'color'.

Active Items with Custom Attributes

The following 4 items were found matching criteria:

ITEM NAME	SKU	STOCK LEVEL	WARRANTY TYPE
Monitor Pro	MON-P01	25	3 Year Extendable
Wireless Mouse	MOUSE-W	150	1 Year Basic
Laptop X9	LAPX-9	80	3 Year Extendable
Webcam HD	CAM-HD	75	N/A

Note: Attributes were pulled using the `list_attributes` *tool.*

U We need to move 10 monitors from the main warehouse to the overflow storage room. Can you log that?

Location Transfer Logged

The following transaction was successfully recorded:

- **Transaction ID:** TXN-20240614-A
- **Item:** Monitor Pro (MON-P01)
- **Quantity Moved:** 10 units
- **From Location:** Main Warehouse Aisle 1
- **To Location:** Overflow Storage Room C
- **Status:** Complete. Stock count updated.

Frequently Asked Questions

01 **How can BoxHero help me track stock levels in real-time?**

BoxHero gives you an accurate view of your current inventory counts by tracking every movement. You don't have to guess—your agent uses the MCP tools to pull live data, telling you exactly what's available right now.

02 **Does BoxHero help me keep track of where my items are in the warehouse?**

Yes. It manages storage locations and tracks item transfers between them. If an item moves from one shelf to another, you can log that movement instantly, keeping your records perfect.

03 **I'm running a small shop; is BoxHero too complex for me?**

No. You interact with it using natural language through your AI agent. It handles the complexity of tracking items and locations behind the scenes, letting you focus on selling.

04 **Can BoxHero generate a full audit trail of stock movements?**

Absolutely. The MCP provides a complete log of every item transaction—every time something was added or removed. This history is crucial for audits and figuring out where inventory discrepancies came from.

05 **What if I need to add custom details like 'color' or 'model number' to an item?**

You can manage those specific details using the MCP. It lets you list available attributes and create new items with all your necessary product metadata included.

06 **Is BoxHero better than just using a big spreadsheet for inventory?**

Yes, because it's live. A spreadsheet is static; BoxHero updates instantly when an item is moved or shipped out. You get real-time data access through your agent, not old numbers from a file.

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 `"boxhero": { "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

BoxHero 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 BoxHero. 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	BoxHero MCP
Server ID	`019d7561-f82a-71ba-9eab-aa4116b0c891`
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/boxhero.

© 2026 Vinkius · vinkius.com · support@vinkius.com