

MCP SERVER

NO CODE

CLOUD HOSTED

# Walmart Marketplace MCP

Manage your entire 3P catalog, inventory, and pricing structure.

Walmart Marketplace MCP manages your entire third-party selling operation. Use it to control catalog items, synchronize live inventory across fulfillment centers, and execute dynamic pricing updates directly through AI agents. It lets you automate bulk price changes, set promotions, and keep your listings accurate without leaving your workflow.

**A+** Quality Score 100/100

marketplace-management

dynamic-pricing

catalog-management

inventory-sync

seller-tools

product-listings



# 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.

# Walmart Marketplace MCP

8 tools available

Cloud-hosted on Vinkius

Managing a large e-commerce catalog is messy work. You need visibility into stock levels, current prices, and item status—all in one place. This MCP gives your agent full control over your Walmart 3P Seller operations using secure API calls. Instead of jumping between seller portals to manually adjust pricing or publish new items, you simply tell your AI client what needs doing. It handles the complexity behind the scenes.

For instance, if you run a seasonal sale, you can use the agent to both set up markdown promotions and update inventory levels simultaneously. The whole system lives on Vinkius, making it one of the most powerful e-commerce connectors available in the catalog. You manage your entire product life cycle—from listing items to retiring them—with direct data access.

---

## Core Capabilities

### 01 — Manage Stock Levels

Automates physical stock allocation across all your fulfillment centers.

### 02 — Bulk Pricing Operations

Applies massive price changes, markups, and cost adjustments across many SKUs at once.

### 03 — Catalog Visibility

Retrieves active item lists and checks the specific publishing status or errors of individual products.

### 04 — Promotion Setup

Creates structured clearance or markdown sale rules for your product lines.

### 05 — Item Lifecycle Control

Publishes new listings, updates existing ones, or completely removes products from the marketplace.

# One Click on Vinkius — From Prompt to Execution

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

- 01** Configure your connection using your Walmart Developer Portal credentials (Client ID and Client Secret) to establish a secure link.
- 02** Select whether you need to run tests in Sandbox or go live in Production mode.
- 03** Your agent executes the desired operation, like updating inventory or setting prices, retrieving confirmation immediately.

The bottom line is that your AI client runs all the complex API interactions so you just focus on the final result: accurate data.

---

## Built For

This MCP is built for e-commerce operators who spend too much time in seller portals. It helps those managing hundreds of SKUs or coordinating rapid, large-scale inventory changes across multiple channels.

### E-Commerce Operations Manager

They use this to coordinate bulk pricing updates and ensure that seasonal promotions are correctly applied to the entire catalog before a major sales event.

### Inventory Planner

Needs to check item status and update inventory levels quickly across different fulfillment centers when stock counts change drastically.

### Catalog Integrator

Uses this MCP to list, retire, and maintain the integrity of product data, ensuring every SKU matches the marketplace requirements.

---

## What Changes When You Connect

- 01** Stop manually adjusting prices across hundreds of SKUs. Use `wm_bulk_price_update` to send price changes for a massive batch of items in one go.

- 
- 02 Maintain real-time stock accuracy. If your physical counts change, use `wm_update_inventory` to sync the data immediately across all locations.

---

  - 03 Keep listings clean and error-free. Before running big campaigns, check item health using `wm_get_item_status` to find out if any SKU has buy-box limits or publishing errors.

---

  - 04 Manage your product lifecycle efficiently. Use `wm_list_items` to get a master list of everything you sell, and then use `wm_retire_item` when an item is permanently discontinued.

---

  - 05 Run major sales quickly. You can set up markdown rules using `wm_setup_promotions`, ensuring your clearance items are priced correctly right away.
- 

---

## Real-World Applications

### Ending a Product Line

A seller needs to pull an item from the market but doesn't want it visible. They ask their agent to run `wm_retire_item` for SKU X, instantly masking it and preventing any customer from finding or buying it.

### Post-Merger Inventory Sync

After acquiring a new warehouse, an operator needs to verify stock. They run `wm_list_items` to see what's active, then use `wm_update_inventory` to accurately allocate the newly received physical stock.

### Coordinating a Flash Sale

A team needs 50 different items on sale. They ask their agent to run `wm_setup_promotions` first, then use `wm_bulk_price_update` to set the final clearance price for all of them.

### Data Cleanup and Audit

A catalog integrator suspects some items are misclassified. They run `wm_get_taxonomy` to get the correct category structure, then use that data to ensure new product listings are correctly placed.

---

## Patterns to Avoid

---

### Updating prices one SKU at a time

#### X AVOID

Jumping into the seller portal and manually changing the price for every single item in a batch sale. This is slow, error-prone, and wastes hours.

#### ✓ INSTEAD

Use ``wm_bulk_price_update`` to send all the necessary price changes at once from your agent. It handles the payload safely.

---

### Forgetting to check item status

#### X AVOID

Running a promotion and only finding out later that half the items were rejected because they had buy-box limits or needed different category codes.

#### ✓ INSTEAD

Always run ``wm_get_item_status`` first. It analyzes specific SKU constraints, giving you an error report before you waste time on bad data.

---

### Assuming inventory is correct

#### X AVOID

Manually updating the listing without confirming stock levels first, resulting in selling items that are actually out of physical stock.

#### ✓ INSTEAD

Check your current count and then use ``wm_update_inventory``. This guarantees the system reflects the actual unit value you have on hand.

---

## The Right Fit

Use this MCP if your core pain point is managing high volume, variable data across many individual listings. If you need to update prices in bulk, manage stock counts at scale, or control an item's entire lifecycle (from listing it to retiring it), this toolset gives you the direct access needed.

Don't use it if all you need is basic read-only reporting on sales history—you might just need a general analytics tool. Also, don't try to use it for competitive intelligence that requires analyzing competitor pricing; you'd be better off using an external market research tool instead.

---

---

## The sheer manual labor of running e-commerce listings is exhausting.

Think about the typical Monday morning. You have to jump into the seller portal, check the inventory tab for Store A, then click over to pricing and make sure all 300 SKUs are priced correctly. If you're running a sale, you repeat that process: set up promotions in one section, manually list prices in another, and if stock dropped overnight, you have to jump back again. It's endless clicking.

With this MCP, your agent handles the coordination. You simply tell it, 'We are launching a clearance sale.' The system then automatically runs `wm_setup_promotions`, verifies item status with `wm_get_item_status`, and can even trigger `wm_bulk_price_update`—all without you ever touching a single portal tab.

---

## Getting Full Control With the Walmart Marketplace MCP

The most painful steps that disappear are the repetitive checks: verifying if an item is ready to sell, finding out why it's stuck in 'draft,' and individually updating stock counts. You don't have to copy data from one spreadsheet into three different seller tabs anymore.

Now, managing your entire catalog feels like giving clear instructions to a trusted teammate. You gain full control over every element of the listing process—from initial setup with `wm_list_items` to final takedown using `wm_retire_item`. It's immediate and complete.

---

# Walmart Marketplace MCP: 8 Tools

Use these specific tools to execute every critical function required for running a third-party marketplace store, from listing data extraction to live stock updates.

#	TOOL	DESCRIPTION
01	<code>wm_bulk_price_update</code>	Updates prices for many items simultaneously using a structured payload feed.
02	<code>wm_get_item_status</code>	Checks the current publishing status, buy-box limits, and any specific errors associated with an SKU.
03	<code>wm_get_taxonomy</code>	Retrieves the correct structural category classification nodes needed for product listing data.
04	<code>wm_list_items</code>	Extracts a clean list of all actively published third-party seller catalog items.
05	<code>wm_retire_item</code>	Securely removes an item from the public marketplace view, masking it completely from customers.
06	<code>wm_set_price</code>	Sets a specific retail price for one or more items according to defined market limits.
07	<code>wm_setup_promotions</code>	Applies clearance, markdown, and promotional rules by submitting structured sale payloads.
08	<code>wm_update_inventory</code>	Changes the physical stock count for an item at a specific fulfillment location.

---

## See It in Action

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

**U** Update the inventory of SKU 8291 to 50 items on fulfillment center 1000.



Successfully updated the inventory quantity using `wm_update_inventory`.

**U** Find the current offer price and publish status of SKU 'LAPTOP-01'.



SKU 'LAPTOP-01' is 'PUBLISHED' with a current active offer price of \$499.00 on the marketplace.

**U** Increase the base price of all ASICS sneakers by 5% effectively immediately.



Bulk price adjustment initiated. 34 ASICS SKUs have been updated with a 5% markup submitted successfully.

---

## Frequently Asked Questions

**01** How do I check if my product listing is ready for sale using the Walmart Marketplace MCP?

You use the `wm_get_item_status` tool. This analyzes the item's lifecycle status and reports on any potential errors or buy-box limitations so you know exactly what needs fixing.

**02** Can I update prices for multiple items at once with `wm_bulk_price_update`?

Yes, `wm_bulk_price_update` is designed specifically to handle large volumes of price adjustments using a structured payload. It's much faster than doing it manually.

**03 What if I need to remove an item from the public listing? Should I use `wm_retire_item`?**

Yes, `wm_retire_item` is the correct tool. It completely removes the item's presence from the marketplace view, making sure it cannot be purchased.

---

**04 Does this MCP help me find the right category structure for a new product?**

You can use `wm_get_taxonomy` to pull safe and dynamic structural node trees. This helps you classify your products correctly before listing them.







---

# 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>"walmart-marketplace": { "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

# Walmart Marketplace 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 Walmart Marketplace. 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	Walmart Marketplace MCP
Server ID	019d761e-a83b-727f-b0ef-e0f4b95c4118
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/walmart-marketplace](https://vinkius.com/mcp/walmart-marketplace).