

MCP SERVER

NO CODE

CLOUD HOSTED

Shopify MCP

Manage your entire online store from natural conversation.

Shopify MCP connects your AI agent directly to your e-commerce backend. You control all aspects of your online store—from managing product inventory and listing custom collections to retrieving detailed customer records and monitoring sales orders—all through natural conversation.

A+ Quality Score 100/100

storefront-management

product-catalog

order-processing

customer-records

multi-channel-commerce

inventory-tracking



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 — connect your AI agent in under 60 seconds.

Shopify MCP

10 tools available

Cloud-hosted on Vinkius

You can run your entire storefront operation from a chat interface. Instead of jumping between multiple tabs or writing complex API calls, you simply ask your AI client what you need. This MCP lets you interact with core Shopify data: check stock levels for specific items, list all managed products, and even find details on past customer purchases.

Need to handle support requests? You can search through customer profiles directly and pull up full order histories for quick troubleshooting. Want a better overview of your catalog? You don't have to look at dozens of spreadsheets; you can ask the agent to list smart collections or retrieve general store metadata. The result is that your AI acts like having a dedicated operations manager sitting right next to you, giving you instant access to every piece of customer and product information.

Core Capabilities

01 — Get specific customer details

You can pull up any client's profile and view their associated metadata.

02 — Track order status and history

The agent retrieves full records for sales orders, letting you know if they were fulfilled or are still pending.

03 — Check product inventory levels

You can retrieve product metadata, including stock counts and details about different variants.

04 — List all store customers

The system pulls a list of your stored clients so you can search for specific names or IDs.

05 — View storefront organization

You can access both manually created and automatically generated product collections.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/mcp/shopify-alternative — connect your AI agent in three steps.

- 01** Subscribe to this MCP and provide your Shopify Shop Name along with the Admin API Access Token from your app settings.
- 02** Connect the MCP to your preferred AI client (Claude, Cursor, etc.).
- 03** Ask your agent a natural language question—for example, 'What's the stock count on T-shirts?' or 'List the last five orders.' The system processes the request and returns actionable data.

The bottom line is that you manage complex e-commerce tasks using simple conversation prompts, keeping all your core store data in one place.

Built For

Anyone whose job involves looking up product details, checking inventory, or helping a customer with an order needs this. It's for the operations specialist who gets sick of switching between Shopify's backend and their CRM to solve one simple support query.

Customer Support Agent

You use it to find a customer by name, retrieve their profile metadata, and check the details of specific orders without leaving your chat window.

E-commerce Manager

You query product information, checking if certain items are low on stock or listing out all managed products for a seasonal promotion.

Operations Specialist

You list recent store orders to coordinate fulfillment with the warehouse team and ensure everything is moving through the pipeline smoothly.

What Changes When You Connect

- 01** Stop jumping tabs. You can list recent store orders and check their details without ever leaving your chat interface, keeping support flow faster than before.

-
- 02** Never guess inventory again. Use the `get_product_details` tool to instantly verify stock counts or metadata for any product variant you need to mention.
-
- 03** Customer service gets a huge boost. You can use `list_customers` and then immediately pull up full details using `get_customer_details`, making personalized responses easy.
-
- 04** Catalog management is simple. You can list all managed products via `list_products` or get an overview of your storefront's organization by listing smart collections.
-
- 05** Operations stay on track. The ability to monitor sales orders and pull customer records means fulfillment coordination happens instantly, right where you work.
-

Real-World Applications

Fulfilling a complex support inquiry

A customer asks why their order is delayed. You use the MCP to first list recent store orders and then drill down using `get_order_details` on that specific ID. This gives you all the necessary proof points (shipping addresses, status updates) instantly for the customer.

Onboarding a new client

You need to check if an old customer exists in your system. You ask the agent to list customers and then use `get_customer_details` on their name, giving you a full history for the sales pitch.

Preparing for a marketing campaign

You need to feature top sellers in a new collection. You ask the agent to list smart collections and then use `list_products` to cross-reference products by vendor, ensuring you don't miss any key items.

Auditing inventory after a sale

A client reports missing stock counts. Instead of checking the physical warehouse, you ask the MCP to list products and then use `get_product_details` on the SKU in question to verify current digital metadata.

Patterns to Avoid

Manually searching for product codes

✗ AVOID

A user remembers a product is 'red shirt' but can't remember the exact SKU or ID number, leading to manual database searches and failed API calls.

✓ INSTEAD

Don't guess. Use `'list_products'` and filter by title or vendor first. Then pass the resulting ID into `'get_product_details'` to get all necessary metadata.

Mixing up order details with customer data

✗ AVOID

A support agent tries to use general store info (`'get_shop_info'`) when they actually need to know if a specific customer was billed correctly on an order.

✓ INSTEAD

If you're talking about billing or fulfillment, always start by listing orders using `'list_orders'`, and then use `'get_order_details'` for the specifics.

Assuming product availability

✗ AVOID

A marketing team plans a promotion based on outdated stock data, resulting in an ad that promises items that are actually out of stock.

✓ INSTEAD

Always confirm real-time inventory. Use `'get_product_details'` to check the current count and variant status before launching any major campaign.

The Right Fit

Use this MCP if your workflow requires constant, conversational access to transactional e-commerce data: customer history, product stock counts, or order statuses. It's designed for agents who need to answer 'what happened?' questions instantly.

Don't use it if you need to perform complex graphic design work, manage external ad campaigns (like Google Ads), or handle financial accounting outside of the core sales ledger. For tasks that involve creating new product images or updating your website theme layout, you'll need a dedicated CMS tool instead. This MCP is about *data retrieval and coordination*, not *creative execution*.

The struggle with siloed e-commerce data

Today, if a customer calls with an issue, you're forced into a three-act play of clicking. You jump from your CRM to check their profile, then over to Shopify's order page to track the shipment, and finally open a third tab just to verify the product metadata or stock status. Copy-pasting IDs across five different screens is exhausting.

With this MCP integrated into Vinkius, all that happens in one conversation. You just ask your agent, 'What's wrong with order #1002?' The system instantly pulls up the customer profile using `get_customer_details`, retrieves the shipment status via `get_order_details`, and even tells you if the item is back-ordered—all without a single tab switch. It's instant, unified control.

Product Data Control with `get_product_details`

Previously, determining product details was slow. You might have had to search through multiple categories just to confirm if a specific variant (like the 'Large' size) still had stock or what its pricing structure was. It was tedious, time-consuming guesswork.

Now, you ask your agent for product information. The system runs `get_product_details` and immediately tells you everything: which variants exist, how many are in stock across all locations, and if the metadata is accurate. You know the facts instantly.

Shopify MCP Catalog: 10 Tools Available

These tools allow you to execute every core e-commerce operation—from product cataloging to order fulfillment tracking—using your AI agent.

#	TOOL	DESCRIPTION
01	<code>create_customer</code>	Registers a new customer account in your Shopify system.
02	<code>get_customer_details</code>	Pulls all available information for a specific existing customer.
03	<code>get_order_details</code>	Retrieves comprehensive details about any single sales order, including line items and status.
04	<code>get_product_details</code>	Fetches all metadata for a specific product, like variants and stock levels.
05	<code>get_shop_info</code>	Gets general information about the entire store setup.
06	<code>list_custom_collections</code>	Lists product collections that you manually created in your storefront.
07	<code>list_customers</code>	Provides a list of all customers currently stored in your database.
08	<code>list_orders</code>	Retrieves a summary list of recent sales orders from the store.
09	<code>list_products</code>	Lists all products available in your catalog, allowing filtering by title or vendor.
10	<code>list_smart_collections</code>	Shows automated product collections that automatically update based on rules.

See It in Action

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

U List the last 5 orders from my Shopify store.



I've retrieved your recent orders. Your last 5 include #1001 (Fulfilled), #1002 (Pending), and three others. Would you like to see the customer details for the pending order?

U Search for a customer named 'John Doe' in Shopify.



Searching customers... I found one match for 'John Doe' (ID: 987654). He has made 3 orders and is a member of your 'VIP' group. Should I retrieve his full order history?

U Show me the details for product ID 123456.



Fetching product info... Product 123456 ('Organic Cotton T-Shirt') has 3 variants (Small, Medium, Large) with a total of 45 units in stock. Would you like to see the pricing for each variant?

Frequently Asked Questions

01 Can I list products using the Shopify MCP?

Yes, you can. The `list_products` tool lets you pull a catalog of all your items and even filter that list by vendor or product title.

02 How do I check if a customer exists with the Shopify MCP?

You use the `list_customers` tool to see an overview of clients. Once you find a name, you can then run `get_customer_details` for their full profile.

03 Can I monitor orders using `list_orders` in Shopify MCP?

Absolutely. Running `list_orders` gives you a summary of your recent sales activity. From that list, you can get deeper insights by calling `get_order_details` on any specific order number.

04 Is the Shopify MCP only for active store owners?

No. Even if you just need to analyze historical data or manage collections, this MCP works. You can use tools like `list_smart_collections` to audit your storefront organization.

05 Which tool should I use for creating a new client?







Use the `create_customer` tool. This registers a brand-new customer into your system, making them visible in your records immediately.

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 <code>"shopify-alternative": { "url": "..."} </code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → 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

Shopify 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 · 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 Shopify. 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	Shopify MCP
Server ID	019dd15b-ddd7-7241-a92a-2d7a451aba96
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/shopify-alternative.