MCP SERVER | NO CODE | CLOUD HOSTED

Bolt MCP for AI Agents

Manage e-commerce transactions and refunds instantly

Bolt lets your AI agent manage everything from one-click checkout to complex refunds. Track every transaction, generate order tokens for secure payments, and check merchant account health—all through natural conversation directly in your workflow.

A+ Quality Score 100/100

one-click-checkout | transaction-management | refund-processing | payment-gateway | conversion-optimization

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

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.

Bolt MCP

10 tools available
Cloud-hosted on Vinkius

Managing e-commerce payments usually means jumping between dashboards: checking status codes, manually looking up customer history, or initiating a refund via a separate form. This MCP changes that. You connect Bolt to any AI client, giving your agent direct access to the full payment lifecycle. Your agent can instantly list recent transactions and pull detailed metadata for any specific charge. Need to fix an order? Your agent handles it—it can capture an authorized transaction or void one if something went wrong before checkout completes. Beyond payments, you can generate secure order tokens to kick off a new purchase session, all without ever leaving your chat interface. It's payment management built into conversation. When you connect Bolt through the Vinkius Catalog, your agent becomes a full-time operations specialist.

Core Capabilities

01 — List Recent Transactions

The agent pulls a list of your most recent payment transactions, giving an overview of activity.

02 — Get Specific Transaction Details

The agent retrieves all available metadata and specifics for one particular transaction ID.

03 — Process Refunds

Your agent can issue partial or full refunds for completed sales, or void authorizations that never went through to capture.

04 — Create Order Tokens

The MCP creates secure order tokens, allowing your agent to initiate a one-click checkout session when needed.

05 — Check Merchant Status

Get real-time data on your merchant account's current operational status and configurations.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your Bolt Private API Key along with the environment (sandbox or live).

02 Your AI client connects, giving it permission to execute payment-related actions against your account.

03 You simply tell your agent what you need—for example, 'Refund $15.00 for transaction ABC-123'—and it executes the required task.

The bottom line is that your AI client talks directly to Bolt, acting like a natural language wrapper around complex financial APIs.

Built For

This MCP is essential for e-commerce operations teams and customer support agents. It's for anyone who spends too much time clicking through separate payment dashboards just to resolve an order issue or process a refund.

Customer Support Specialist

Using the agent, they can instantly look up transaction details and initiate refunds based only on the customer's conversation history.

E-commerce Operations Manager

They use this MCP to monitor webhooks or verify API integrations, ensuring payment flows run smoothly without manual dashboard checks.

Developer / QA Tester

Developers test and validate order token generation or specific transaction data retrieval using natural language prompts for quality assurance.

What Changes When You Connect

01 Stop switching between payment dashboards. Your agent handles transaction oversight, letting you list recent payments or pull deep metadata on a single charge without leaving your chat.

02 Streamline customer support resolution. Instead of copying reference numbers, the agent gets order details and transaction histories directly from their workflow tools.

03 Secure checkout is simple. Use the `create_order_token` tool to generate payment tokens instantly, making one-click checkouts easy for customers and your team.

04 Control payments end-to-end. If a sale fails or needs adjustment, you can use `void_transaction` on an authorization or execute a refund with `refund_transaction` —all in conversation.

05 Stay compliant and operational. You can quickly check the merchant account's status using `get_merchant_status`, verifying everything is running right before a big sale.

Real-World Applications

A customer claims they were double-charged.

The agent uses `list_transactions` to show the customer all recent charges. They then use `get_transaction` on the relevant entry, confirming a duplicate payment and automatically initiating a refund via `refund_transaction`.

An employee needs to check if an order was cancelled.

The agent fetches the specific order details using `get_order_details`. If the transaction is pending, they can use `void_transaction`, confirming the cancellation without needing internal API access.

A developer needs to test a payment flow.

The agent uses `create_order_token` to simulate a new purchase session. They then call `get_merchant_status` to verify that the sandbox environment is correctly configured for testing.

A team needs to audit webhooks.

The agent calls `list_webhooks`, providing an instant, auditable list of every real-time data sync endpoint connected to the merchant account.

Patterns to Avoid

Trying to debug refunds manually

✗ AVOID

Copying and pasting transaction IDs into a separate dashboard or calling multiple internal APIs in sequence. This is slow, prone to human error, and requires constant context switching.

✓ INSTEAD

Let the agent handle it. Use `refund_transaction` directly in conversation after confirming the correct details with `get_transaction`. The whole process stays contained.

Forgetting to check payment status

✗ AVOID

Attempting to capture funds on a transaction that was already voided, resulting in an API failure and requiring manual investigation.

✓ INSTEAD

Always run `get_merchant_status` first. This confirms if the account is ready for payments. If unsure, use `list_transactions` to verify the current status of the payment lifecycle.

Misidentifying order scope

✗ AVOID

Asking the agent about a customer's general history when you only need data on a specific purchase attempt, leading to irrelevant or incomplete information.

✓ INSTEAD

Always start by using `get_order_details` with the specific order ID. This narrows the focus and gives the agent the precise context needed for accurate action.

The Right Fit

Use this MCP if your main bottleneck is resolving payment disputes or managing refunds across multiple systems. If you need to programmatically manage payments, refund amounts via `refund_transaction`, or verify checkout status using `create_order_token`, this tool is critical.

Don't use it if all you want to do is look at a simple CSV of sales figures—a basic reporting tool handles that better. Also, don't rely on it for non-financial tasks like inventory tracking or user profile management; those require different MCPs. When in doubt about payment status, always check the `get_merchant_status` first.

Bolt MCP: Managing E-commerce Payment Refunds with AI Agents

Today, processing refunds means a tedious dance of clicks. You have to navigate to the transaction history dashboard, search for the specific order reference number, verify the original amount, and then manually input details into a separate refund form. It's slow, it requires multiple logins, and you constantly risk miskeying an ID or referencing the wrong payment cycle.

With this MCP, that entire manual flow disappears. Your agent takes the customer's request—'I need a partial refund for order XYZ'—and executes the necessary steps internally. It pulls the details with `get_transaction` and issues the credit using `refund_transaction`. You get confirmed outcomes in real time.

Bolt MCP: Automating Checkout and Order Token Generation

Before this, setting up a one-click checkout required developers to write dedicated code paths just to generate secure tokens. Operations teams had to manually verify that the API credentials

Now, your agent handles it. You can trigger `create_order_token` in natural language to initiate a payment session or check account readiness using `get_account_details`. It’s reliable, secure, and ready the checkout process instantly.

were correct before every major sale, spending hours on setup validation.

Bolt: 10 Tools for Transaction & Order Management

These tools allow your agent to perform every key action in the payment lifecycle, from viewing history to issuing refunds.

#	TOOL	DESCRIPTION
01	`capture_transaction`	Finalizes a pending payment by capturing the funds from an already authorized transaction.
02	`create_order_token`	Generates a unique, secure token required to start a one-click checkout session for a customer.
03	`get_account_details`	Retrieves the general and current configuration details of your merchant account.
04	`get_merchant_status`	Checks the real-time operational status, ensuring your merchant account is active and configured correctly.
05	`get_order_details`	Fetches comprehensive details for a specific order associated with a customer or reference number.
06	`get_transaction`	Retrieves all available metadata and specifics for one particular transaction ID.
07	`list_transactions`	Pulls a list of your most recent payment transactions, giving an overview of activity.
08	`list_webhooks`	Audits and lists all currently configured webhooks to ensure real-time data sync is working correctly.
09	`refund_transaction`	Initiates a refund, which can be full or partial, for any completed transaction ID.
10	`void_transaction`	Reverses an authorized payment that failed to complete and needs to be cancelled before capture.

See It in Action

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

U List my last 5 transactions and tell me if any are authorizations that need voiding.

Recent Transaction Summary

REF ID	TYPE	AMOUNT	STATUS
ABC-123	Charge	$45.00	Completed
XYZ-789	Auth	$120.00	Pending $
QRS-444	Charge	$22.50	Completed

The authorization (XYZ-789) is still pending and ready to be voided if the order was cancelled.

U I need a partial refund of $10 for transaction ABC-123.

Refund initiated.

We've successfully started a partial credit of **$10.00** (1000 cents) for the original payment reference ABC-123. The funds should appear in your account within 3-5 business days.

U What is the current status of my merchant account?

Merchant Account Status Report

- **Status:** Active and Operational
- **Publishable Key Endings:** ...xyz
- **Division ID:** div_123
- **Last Checked:** 2026-06-12. All configurations are valid for immediate payment processing.

Frequently Asked Questions

01 **How can I use the Bolt MCP to process a refund?**

You simply ask your agent to issue a refund and provide the transaction reference or order ID. The agent handles checking if the payment is complete, verifying the details, and executing the refund through the necessary tools.

02 **Can I use this MCP for payments from different countries?**

Yes, as long as your Bolt Merchant account supports those currencies and transaction types. The MCP functions on whatever payment flows are configured within your connected merchant account.

03 **What if my order fails before checkout? Can I cancel the authorization?**

Absolutely. You can use the MCP to void pending authorizations using tools like `void_transaction`. This reverses the temporary hold on funds, preventing issues later in your accounting cycle.

04 **Is the Bolt MCP only for viewing data?**

No. It's a full management tool. You can actively perform actions like creating order tokens, capturing authorized payments, and issuing refunds right from your chat interface.

05 **Do I need to manually update my API keys in the MCP?**

Once you connect it via Vinkius, you enter your private API key once. The connection persists so your agent can access payment data and execute commands whenever needed.

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 `"bolt": { "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

Bolt 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 Bolt. 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	Bolt MCP
Server ID	`019d7560-a1c6-7329-9341-1680d032e20a`
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/bolt.

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