MCP SERVER | NO CODE | CLOUD HOSTED

Afterpay MCP for AI Agents

Manage Buy Now, Pay Later checkouts and refunds

Afterpay manages Buy Now, Pay Later transactions by giving your agent direct access to checkout, payments, and refunds. You can use this MCP to initiate secure payment sessions for customers, audit historical authorizations, track capture statuses, or process full/partial refunds directly from a chat interface.

B Quality Score 87.3/100

buy-now-pay-later | checkout-sessions | refund-processing | payment-status | merchant-services | financial-transactions

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

SCAN TO VIEW

The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect 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.

Afterpay MCP

5 tools available
Cloud-hosted on Vinkius

This connector gives your AI client the ability to handle complex financial transactions related to Afterpay (Clearpay). Instead of logging into a merchant portal and clicking through multiple tabs, your agent manages the entire payment lifecycle conversationally. You can start by initiating new checkout sessions for customers, generating secure tokens on demand. When things are already happening, you can ask your agent to list all historical payments or check the detailed status of any specific order ID. Need to fix a billing issue? Your agent handles full or partial refunds immediately. All these capabilities—from checking minimum and maximum order limits to monitoring capture statuses—are routed through the Vinkius catalog, letting you manage everything without leaving your chat window.

Core Capabilities

01 — Initiate secure checkout sessions

Your agent creates a new payment session token for a customer transaction amount.

02 — Audit historical payments and authorizations

You retrieve comprehensive lists of past BNPL transactions, including authorization records and technical metadata.

03 — Retrieve detailed order financial status

The agent pulls full settlement information, payment statuses, and transaction logs for a specific Afterpay order ID.

04 — Execute refunds on captured orders

You initiate immediate full or partial refunds against previously authorized payments.

05 — Verify merchant configuration limits

The agent confirms the minimum and maximum transaction amount rules for your Afterpay account.

One Click on Vinkius — From Prompt to Execution

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

01 First, subscribe to this MCP and enter your specific Afterpay Merchant ID and Secret Key into your AI client.

02 Next, tell your agent exactly what you need—for example, 'Check the status for order 12345' or 'Create a new checkout for $50'.

03 Your agent executes the necessary API calls using the provided credentials and returns real-time financial data or confirmation of the action taken.

The bottom line is, you connect your Afterpay merchant account once, then use natural language to run complex payment operations through any MCP-compatible client.

Built For

Finance Ops teams and E-commerce Managers need this. It's for anyone who spends too much time switching between the merchant dashboard and internal ticketing systems just to verify a refund or check an order limit. If you live in the details of payment flow, this is your MCP.

Finance Operations Analyst

They use it to automate audit processes, quickly listing historical payments and executing partial refunds based on support tickets.

E-commerce Manager

They check checkout health by verifying order limit configurations or creating new payment sessions for marketing promotions instantly.

Customer Support Specialist

They look up specific payment statuses and comprehensive order history to resolve billing inquiries without escalating to a supervisor.

What Changes When You Connect

01 Stop manual dashboard exports. You can quickly identify successful captures or pending authorizations using the transaction insights capability.

02 Handle refund processing instantly. Use the `refund_payment` tool to initiate partial or full credits directly through conversation.

03 Monitor checkout health on the fly. Verify order limit configurations or check minimum/maximum limits with `get_afterpay_config` before launching a sale.

04 Saves time in support tickets. Instead of guessing, you use the agent to retrieve detailed financial status and logs for any customer order ID.

05 Streamlines payments. The `create_checkout` tool allows your agent to generate secure payment session tokens immediately when needed.

Real-World Applications

Investigating a disputed charge

A support specialist needs to check why a customer was charged incorrectly. They ask the agent to look up the order ID, using `get_payment_details`. The agent returns the exact settlement logs, allowing the specialist to confirm if the payment status is truly 'APPROVED' or pending.

Running a flash sale with new limits

The e-commerce manager needs to know if their planned promotion amount exceeds current rules. They ask the agent for configuration details, calling `get_afterpay_config`, ensuring the transaction falls within established minimum and maximum order limits.

Processing an urgent customer return

A team member processes a refund immediately after confirming the original capture. They ask the agent to run `refund_payment` for the specific order ID, completing the financial reversal in minutes instead of hours.

Auditing quarterly transaction volume

The finance analyst needs a clean list of all payments made last month. They ask the agent to run `list_payments`, getting a secure summary of all historical BNPL transactions for immediate review and reporting.

Patterns to Avoid

Manual data extraction

✗ AVOID

Having to log into the Afterpay dashboard, manually filter by date range, and copy/paste transaction statuses into a spreadsheet.

✓ INSTEAD

Ask your agent to `list_payments` or use `get_payment_details`. The tool pulls all necessary records directly, providing structured data in seconds.

Incorrect refund amounts

✗ AVOID

Assuming the correct partial refund amount without checking if previous payments were fully captured or if there are balance restrictions.

✓ INSTEAD

Always run `get_payment_details` first. This verifies the remaining balance and confirms you can safely initiate a refund using `refund_payment`.

Ignoring transaction limits

✗ AVOID

Attempting to create a checkout session that exceeds the maximum allowed order amount, which results in an immediate payment failure.

✓ INSTEAD

Check your rules first. Run `get_afterpay_config` to verify minimum and maximum order amounts before calling `create_checkout`.

The Right Fit

Use this MCP if your core job involves managing the operational flow of Buy Now, Pay Later payments—specifically when you need to initiate checkouts, track historical transactions, or process refunds. It’s perfect for finance ops teams and support staff who deal with payment status changes daily.

Don't use it if you just need a high-level financial overview that doesn't require specific action (like running reports on all payments). For pure reporting without interaction, an export tool might be better. But if the goal is to *act*—to refund money or create a new checkout session—this MCP provides the necessary controls via `refund_payment` and `create_checkout`. It's your conversational command center for Afterpay finance.

Afterpay MCP: Automating BNPL Payment Status Checks with AI

Today, managing payment status means jumping between the merchant dashboard and internal tools. You have to click through order IDs, check settlement reports, and cross-reference multiple tabs just to confirm if a payment was captured or if a refund is eligible. It’s slow and prone to manual errors.

With this MCP, you simply ask your agent for the details using `get_payment_details`. You get an instant summary of the financial status—everything from settlement info to logs—right in your chat window. That's verifiable payment truth, instantly.

Afterpay MCP: Managing Checkout Sessions and Order Limits via AI

Before running a sale or creating a new checkout link, you typically have to manually check the Afterpay merchant portal to confirm the minimum

Now, just ask your agent for configuration details using `get_afterpay_config`. You'll instantly know the hard rules—the min/max amounts—before

and maximum order limits. If you get this wrong, the transaction fails immediately.

generating any payment session token with `create_checkout`. It eliminates guesswork entirely.

5 Afterpay Tools for Managing Payments and Refunds

Use these tools to create checkout sessions, list historical transactions, get detailed status reports, or execute full/partial refunds immediately.

#	TOOL	DESCRIPTION
01	`get_afterpay_config`	Retrieves the minimum and maximum transaction limits set on your Afterpay merchant account.
02	`create_checkout`	Starts a secure payment session token for a customer's Buy Now, Pay Later purchase.
03	`list_payments`	Gathers a list of all historical BNPL transactions and authorizations from your account.
04	`get_payment_details`	Pulls detailed financial status, settlement information, and logs for one specific Afterpay order ID.
05	`refund_payment`	Processes a full or partial refund to credit a customer against a previously captured payment.

See It in Action

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

U I need to process a refund for order ID 89012. Can you check the details first?

Order Details: 89012

- **Status:** Captured (3/15/2024)
- **Amount:** $199.99
- **Refundable Balance:** Yes, full refund is available.

Please confirm if you'd like to process a partial or full credit.

U What are our current Afterpay limits? I'm planning a big sale.

Afterpay Merchant Configuration

METRIC	LIMIT
Min Order Amount	$1.00
Max Transaction Limit	$2,500.00

These limits apply to all US and Canada transactions.

U List payments for the last 3 weeks so I can audit our sales.

Historical Payments Summary

- Total Transactions Found: 45
- Successful Captures (Last 7 Days): 28
- Pending Authorizations: 17

Please specify if you want to filter by status or date range.

Frequently Asked Questions

01 **How does the Afterpay MCP help with refund processing?**

The Afterpay MCP allows your agent to handle refunds directly. You simply provide the order ID, and the tool verifies the remaining balance before initiating a full or partial credit on the customer's account.

02 **I need to know if I can create new checkout links using the Afterpay MCP?**

Yes, you can use the MCP to generate secure payment session tokens. Just tell your agent the amount, and it handles creating the necessary redirect URL for your customer.

03 **What kind of payments or transactions can I check with this MCP?**

You can audit all types of BNPL activity, including historical authorizations, detailed settlement information, and current payment statuses. It covers everything from initial checkout to final refund.

04 **Is there a way to check my Afterpay order limits before selling something?**

Absolutely. You use the MCP to retrieve your merchant configuration, confirming the minimum and maximum amounts allowed per transaction. This prevents failed payments due to policy violations.

05 **Does the Afterpay MCP work with different AI clients like Claude or Cursor?**

Yes. As long as your client supports the Model Context Protocol (MCP), you can connect it. You access all the payment tools through your preferred agent interface.

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 `"afterpay": { "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

Afterpay 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 Afterpay. 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	Afterpay MCP
Server ID	`019d7549-2916-7255-8638-1c5dd9c2d590`
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/afterpay.