

MCP SERVER

NO CODE

CLOUD HOSTED

FastSpring MCP

Control your payments and subscriptions via AI.

FastSpring lets your agent take full control of global digital commerce, payments, and subscriptions. Track orders, manage billing rules, check account details, and handle entire subscription lifecycles—all through natural conversation. It gives you direct access to complex payment data usually locked in a dashboard, letting your AI client perform deep audits or trigger changes without manual navigation.

A+ Quality Score 100/100

merchant-of-record

global-tax-compliance

subscription-management

digital-commerce

order-processing

fraud-prevention



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.

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.

FastSpring MCP

10 tools available

Cloud-hosted on Vinkius

Need to audit a customer's transaction history or manage recurring billing? This MCP connects your agent directly into the FastSpring platform, giving it full control over global payments and subscription management. Instead of logging into multiple dashboards and manually searching through logs, you can talk to your AI client and get answers immediately. You can check specific order details, verify if a customer successfully passed fraud checks, or even handle complex plan updates for existing subscriptions.

It's built for deep operational work. You can use it to identify and update core customer records, ensuring that profile data stays current across your system. Need to cancel a subscription? The agent handles the process securely while pulling out rich notes about why the customer left—that's valuable churn metadata. This level of payment orchestration is usually hidden away deep in a platform backend; Vinkius makes it available via natural language commands, turning complex commerce operations into simple conversations.

Core Capabilities

01 — Audit Order History

The agent retrieves detailed cloud logs to trace order limits and confirm if fraud filters passed for specific transactions.

03 — Update Customer Accounts

The agent finds and updates customer records within the system using profile data like email addresses.

05 — Verify Product Catalog

The agent retrieves a list of configured packages and confirms which digital products are currently active in your store.

02 — Manage Subscriptions

You can check renewal dates, apply currency rules, or trigger ad-hoc billing charges on managed plans.

04 — Cancel Subscriptions

It securely terminates subscriptions, extracting valuable notes about why the customer is leaving (churn reasons).

One Click on Vinkius — From Prompt to Execution

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

- 01** Subscribe to this MCP and provide your FastSpring API Username and Password, found in the Dashboard under Integrations.
- 02** Connect your AI client. Your agent now has access to all commerce tools via natural language prompts.
- 03** Ask a question or give a command—like 'Check order ORD-12345'—and the MCP executes the necessary API calls.

The bottom line is, you use your AI client to talk directly to FastSpring's operational backend without ever touching a dashboard.

Built For

This tool solves the problem for operations and engineering teams who are tired of clicking through dozens of dashboards just to answer simple customer questions or run financial audits. It's essential for anyone whose job involves payment logic, billing cycles, or managing SaaS subscriptions.

Customer Success Manager

You use the agent to look up specific order details and instantly generate time-limited authentication links when a customer calls asking about their account.

SaaS Operations Engineer

You audit product catalogs or test complex subscription proration logic, confirming billing rules across different currency applications without writing boilerplate code.

Finance Analyst

You execute bulk processes, like manually triggering renewals or managing revenue arrays to reconcile month-end financial reports quickly.

What Changes When You Connect

-
- 01 Audit complete order history using `get_order_details`. You don't just see the status; you get cloud logs confirming if fraud filters passed, which is critical for compliance checks.

 - 02 Handle customer accounts with precision. Use `list_accounts` to find all user records and `update_account_info` when a client changes their email or name.

 - 03 Stop guesswork on billing cycles. The agent uses `get_subscription_details` to inspect internal arrays, letting you verify complex plan math for proration or renewals.

 - 04 Keep your customers happy with quick access. Running `generate_auth_link` gives Customer Success the ability to instantly issue secure login URLs without manual workarounds.

 - 05 Manage churn systematically. Instead of just cancelling an account, use `cancel_subscription` which pulls rich metadata detailing *why* the customer is leaving.
-

Real-World Applications

Troubleshooting a Failed Order

A user needs to know why order 'ORD-90210' failed. Instead of checking five different internal dashboards, they ask their agent to use ``get_order_details``. The agent immediately confirms the exact failure point and if fraud filtering was the issue.

Reconciling Old Revenue

The finance team needs to manually trigger renewals for a specific cohort of users. They command their agent to run bulk iterations, executing the revenue recovery process by calling tools like ``charge_managed_subscription``.

Mid-Cycle Plan Upgrade

A client calls wanting to upgrade their plan. The operations team asks their agent to first check current details using ``get_subscription_details`` and then apply the change safely with ``update_subscription_plan``, all in one conversation.

Mass Account Cleanup

A manager discovers several outdated user accounts. Using ``list_accounts``, they find all stale records and then use ``update_account_info`` to mass-clean the profile data, keeping the system accurate.

Patterns to Avoid

Assuming basic info is enough

✗ AVOID

A user asks: 'What's the status of this subscription?' They only get a simple 'Active.' which doesn't explain billing rules or renewal dates.

✓ INSTEAD

To understand the full picture, ask the agent to use ``get_subscription_details``. This tool inspects deep internal arrays and provides precise plan math details you need.

Trying to manually reset a link

✗ AVOID

A customer can't log in and the support rep has no idea how to generate a temporary access token, forcing them to wait until IT is available.

✓ INSTEAD

Ask your agent to use ``generate_auth_link``. It automatically runs an automated validation check and provides a working, time-limited URL for immediate customer service resolution.

Using the wrong tool for updates

✗ AVOID

A developer tries to update billing rates by manually editing a record in the database instead of using the proper API endpoint.

✓ INSTEAD

Always use ``update_subscription_plan`` or ``get_account_details``. These tools ensure you are following FastSpring's structured rules when changing plans or customer data.

The Right Fit

Use this MCP if your workflow requires deep, programmatic access to payment logic, subscription life cycles, and global billing records. You need more than just a simple 'read order status'; you need to audit fraud passing rates (`get_order_details`), update complex user profiles (`update_account_info`), or calculate precise proration charges (`charge_managed_subscription`). Don't use this if your only goal is checking general website content or retrieving non-payment related customer information. If the task involves money, billing rules, plans, or account ownership, you need FastSpring.

The Pain of Manual Billing Audits

Right now, checking a single client's payment history means jumping between multiple tabs and dashboards. You pull up the order ID in one system, check fraud logs in another, then you have to manually look at the billing tab for renewal dates and plan details. It's slow, prone to copy-paste errors, and leaves you with a mess of screenshots instead of an answer.

With this MCP, you just talk to your agent. You ask it to audit the order, and it pulls all that data together—the logs, the billing rules, the plan math—and gives you one clean, accurate summary. The result is instant operational clarity.

FastSpring MCP: Instant Control Over Billing Data

You no longer have to wait for a billing specialist to run reports on churn reasons or verify if a specific customer passed fraud filtering. You can ask the agent to use `cancel_subscription`, and it doesn't just shut the account down; it captures rich metadata explaining *why* the cancellation happened.

What changes is that your entire team moves from reactive, manual data gathering to proactive, conversational operations. The billing truth is available whenever you need it.

FastSpring: 10 Tools for Commerce Ops

These tools let your agent perform specific actions within FastSpring, covering everything from retrieving order logs to updating customer profiles.

#	TOOL	DESCRIPTION
01	<code>cancel_subscription</code>	Terminates a managed subscription and returns detailed notes about the reason for cancellation.
02	<code>charge_managed_subscription</code>	Applies structured billing rules to calculate and export active charges for a managed plan.
03	<code>get_account_details</code>	Extracts core properties needed to check or update an existing customer account record.
04	<code>generate_auth_link</code>	Creates a temporary, limited-time login link for a user to access their private portal.
05	<code>get_order_details</code>	Retrieves comprehensive cloud logs detailing an order's status and confirming if fraud filtering was successful.
06	<code>get_subscription_details</code>	Inspects deep internal data arrays to understand how specific billing plans are calculated or measured.
07	<code>list_accounts</code>	Identifies and lists all existing customer records housed within the FastSpring platform.
08	<code>list_catalog_products</code>	Retrieves a complete list of configured products, verifying their mapping status in your store.
09	<code>update_account_info</code>	Processes and saves updated customer profile information by generating a new JSON payload.
10	<code>update_subscription_plan</code>	Changes an active subscription to reflect a different plan type, maintaining accurate billing records.

See It in Action

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

U What is the status of FastSpring order 'ORD-12345'?



Retrieving order details... Order ORD-12345 is currently 'Completed'. The customer successfully passed fraud filtering and the license keys have been delivered. Would you like the buyer's email address?

U Generate a 24h auth link for account 'acc_abc123'



Auth link generated! Here is the ephemeral login URL for account acc_abc123: <https://yourstore.onfastspring.com/authenticate/jwt...> This link will expire in 24 hours.

U Cancel subscription 'sub_xyz789' and tell me why



Subscription cancelled! sub_xyz789 has been terminated. The recorded churn metadata indicates the reason was 'Price too high'. The customer will no longer be charged for this plan.

Frequently Asked Questions

01 How do I check an order status using FastSpring MCP?

You use the `get_order_details` tool. This retrieves explicit cloud logs that show not only the order status but also if the customer successfully passed fraud filtering.

02 Can I change a subscription plan with FastSpring MCP?

Yes, you use the `update_subscription_plan` tool. This lets your agent identify precise active arrays and update the user's billing structure accurately.

03 What is needed to access FastSpring MCP?

You must subscribe to this MCP on Vinkius and provide your FastSpring API Username and Password from your dashboard credentials area.

04 Does FastSpring MCP handle customer data updates?

Yes. You can use `update_account_info` to provision a new JSON Payload, allowing you to update core customer details like email addresses across the platform.

05 How do I generate an authentication link using FastSpring MCP?







Use the `generate_auth_link` tool. It runs an automated validation check and instantly creates a secure, time-limited login URL for your customer service team to use.

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>"fastspring": { "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

FastSpring 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 FastSpring. 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	FastSpring MCP
Server ID	019d7596-f825-72d3-b8b6-48963bf9d1f3
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/fastspring.