MCP SERVER | NO CODE | CLOUD HOSTED

Increase MCP

Execute real bank actions via your AI agent.

The Increase MCP gives your AI agent direct, programmatic access to a fully compliant commercial US bank. You can instantly build new live accounts, generate required routing numbers, fetch real-time ledger balances, and execute high-value ACH or wire transfers directly from an API call.

A+ Quality Score 100/100

commercial-banking | payment-rails | ledger-management | api-banking | compliance | wire-transfers

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

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.

Increase MCP

12 tools available
Cloud-hosted on Vinkius

This connector lets your agent perform actual banking actions. It connects your AI client to the infrastructure of a physical US bank, letting it handle complex financial transactions that used to require manual dashboard navigation and compliance checks. Need to inject funds into a supplier? Your agent handles the ACH transfer. Want to check if an account is ready for spending? You pull the live ledger balance. The system even includes sandbox arrays, so you can run full simulations—like receiving $1,000 in inbound funds—before touching production money. When your AI client connects through Vinkius, it gets a single point of access to this entire suite of financial tools, letting agents act like true treasury operations staff.

Core Capabilities

01 — Create new bank accounts

The agent provisions completely new live sub-ledger bank accounts on demand for specific internal tracking.

02 — Process outbound payments

Your AI client sends compliant funds using both ACH transfers and high-value, same-day wire services to any US bank.

03 — Manage banking structure data

The agent generates necessary ABA routing numbers and corresponding account identifiers for new financial contexts.

04 — Check ledger status and history

You pull real-time balances or retrieve the full transaction history for any specific account you own.

05 — Test payment flows safely

The agent runs full simulations, such as simulating inbound ACH or wire deposits, without moving a single dollar in production.

One Click on Vinkius — From Prompt to Execution

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

01 First, you decide if your agent needs to operate in Development (Sandbox) mode for testing or Live (Production) mode.

02 Next, you obtain the necessary API Key from Increase and plug that key directly into Vinkius. This syncs the core banking network with your AI client's logic.

03 Finally, your agent calls the required tool—whether it's fetching a balance or creating an account—and gets immediate confirmation of the action taken.

The bottom line is that you give your AI client a direct, compliant financial connection point to execute banking operations on demand.

Built For

This MCP is essential for FinTech builders and corporate finance ops teams who are tired of moving money or checking compliance details manually. It's for the systems engineer who needs to build a fully automated payment rail, or the treasury manager who needs real-time balance checks across multiple virtual accounts.

Platform Builder (BaaS)

They use this MCP to programmatically build internal software that handles money mapping and account provisioning for their own users.

Treasury Analyst

They run full audits, checking historical transactions or simulating fund inflows to predict cash flow without risking actual capital.

Operations Engineer

They automate the setup of new payment channels by requesting fresh ABA routing numbers and creating dedicated bank accounts for temporary operations.

What Changes When You Connect

01 Instant Account Provisioning: Forget waiting days for banking setup. Use `increase_create_account` to provision a live, usable sub-ledger account in minutes for specific internal tracking purposes.

02 Compliance-Grade Payments: When you need to move money, your agent executes it compliantly. Use `increase_create_ach` or the high-value `increase_create_wire` tool to handle payments securely across US payment rails.

03 Financial Structure Control: You gain control over the basic data needed for any financial system. The `increase_create_routing_number` tool generates fresh, usable ABA routing numbers and account identifiers on demand.

04 Operational Visibility: Need to know where your money is? Use `increase_get_balance` to pull a real-time ledger balance or run `increase_list_transactions` to build a complete audit trail for any specific account.

05 Risk-Free Testing: You can validate complex financial logic using the sandbox tools. Run simulations like `increase_simulate_inbound_ach` before deploying code that handles actual funds.

06 Asset Management: The system lets your agent manage associated assets, allowing you to issue and track virtual or physical cards via `increase_create_card` and `increase_list_cards`.

Real-World Applications

Onboarding a New Client Payment Rail

A platform builder needs to onboard a new client that requires dedicated funding. Instead of manual paperwork, the agent first uses `increase_create_account` to generate a unique account ID. Then, it calls `increase_create_routing_number` to get the necessary ABA data, providing immediate readiness for payments.

Pre-flight Payment Audit

A treasury analyst is updating payment software and needs to test fund reception. They run an agent command using `increase_simulate_inbound_ach` against the sandbox environment, confirming that their webhooks correctly process a simulated $5,000 deposit before going live.

Automated Expense Reporting

An operations engineer builds a payment workflow that must trigger both an ACH transfer and issue a virtual card. The agent first uses `increase_get_balance` to check the available funds, then executes `increase_create_ach`, and finally calls `increase_create_card` to cap the spending.

Investigating Past Funds Movement

A compliance officer needs to audit a client's activity over Q2. The agent pulls all records by calling `increase_list_transactions`, allowing them to reconstruct the entire financial history from one centralized tool call.

Patterns to Avoid

Assuming Data is Always Available

✗ AVOID

The developer tries to write payment logic that assumes a valid account number and routing number exist in the database. This fails at runtime because they haven't generated them yet.

✓ INSTEAD

Always use `increase_create_routing_number` first to generate compliant ABA data, then use `increase_create_account` with that data before attempting any transfers.

Testing Live Transactions

✗ AVOID

A developer runs payment simulations directly against the production API key in a hurry. This risks accidentally moving real money or triggering unnecessary fraud alerts.

✓ INSTEAD

Never test live transactions until you're ready to deploy. Always run your initial checks using `increase_simulate_inbound_ach` or `increase_simulate_inbound_wire` first.

Ignoring Transaction History

✗ AVOID

The agent completes a payment but the user has no way to verify that the transaction was actually booked and recorded in the ledger.

✓ INSTEAD

After any transfer, always call `increase_list_transactions` or `increase_list_transfers` immediately afterward. This confirms the record exists and provides proof of operation.

The Right Fit

Use this MCP if your workflow requires making real-world financial moves—creating accounts, sending money that matters, or checking balances against a live ledger. You need compliant US banking infrastructure access. Don't use it if you just need general data points, like reading a CSV file or querying internal operational records. If you only need to *read* historical logs without the ability to initiate transfers, consider an MCP focused purely on document retrieval; this one is about execution and compliance.

Handling Payments Feels Like Doing Accounting in Excel

Today's process for managing funds is a mess of tabs. You check the balance via dashboard A, copy the routing number into form B, then manually enter the payment details on platform C. If anything fails, you're stuck refreshing dashboards and cross-referencing spreadsheets to find out why.

With this MCP, your agent handles the whole chain in one go. Instead of clicking through multiple platforms and copying numbers, you tell your agent: 'Pay $X from Account Y to Z.' It runs all the necessary checks—balance, routing, compliance—and executes it cleanly.

Increase MCP Provides Full Control Over Your Financial Plumbing

The painful steps that disappear are account setup and payment initiation. You no longer need to wait for manual approvals or dedicated infrastructure teams just to get a routing number; you generate one with `increase_create_routing_number` on demand.

This changes everything. Your agent doesn't just report data; it becomes the fully compliant, programmatic financial arm of your business.

Increase MCP with 12 Tools

Use these tools to automate every aspect of your financial workflow—from creating new bank accounts to executing complex payment transfers.

#	TOOL	DESCRIPTION
01	`increase_create_account`	This tool spins up a brand-new, live sub-ledger bank account using programmatic commands.
02	`increase_create_ach`	It sends an outbound ACH transfer of funds to any US bank recipient.
03	`increase_create_card`	This issues both physical and virtual debit cards linked directly to a specific account.
04	`increase_create_routing_number`	It generates new, unique ABA routing numbers and associated account data.
05	`increase_create_wire`	The agent executes a same-day US Wire transfer for large or urgent sums of money.
06	`increase_get_balance`	You fetch the current, real-time ledger balance for any specified account ID.
07	`increase_list_accounts`	The agent can list all sub-accounts that exist under your established charter number.
08	`increase_list_cards`	It sweeps and lists all active debit cards that have been previously issued by the system.
09	`increase_list_transactions`	This retrieves a full, booked history of financial transactions for audit purposes.
10	`increase_list_transfers`	The agent can pull an auditable list of all outbound transfers that have occurred.
11	`increase_simulate_inbound_ach`	This runs a sandbox simulation to model receiving ACH inbound funds without moving real money.
12	`increase_simulate_inbound_wire`	It simulates the receipt of a Wire transfer in the testing environment, perfect for pre-deployment checks.

See It in Action

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

U Use our main test routing sandbox mechanism to simulate inbound an external payload of $1000 into Account ID 'acc_1234'.

Simulation dispatched! Expected parameters were simulated into the core routing infrastructure successfully. The endpoint responded and acknowledged $1,000 artificial liquidity injection into your routing destination target. Validate your app webhooks now.

U Audit our entire open accounts layout right now.

Found exactly 4 Demand Deposit accounts running 'opened' conditions. They map respectively under entity groups 'Operational', 'Cloud Subscriptions', 'Holding', and 'Local Dev'. Current aggregate counts reflect stability.

U Spin up a new fresh physical corporate banking account dedicated uniquely to 'Server Spends'. Send the Routing number to me.

Success chain completed in sequence. First, entity 'Server Spends' received a booked ID structure constraint within banking logic. Secondly, the routing generator hooked an incoming ABA/Account array assigned completely to this context. Account numbers are currently available.

Frequently Asked Questions

01 **Can I use Increase MCP to check my current balance?**

Yes, you can check balances using `increase_get_balance`. This tool fetches the real-time ledger status for any specific account ID within the system.

02 **Is increase_create_wire safe to use in production?**

The transfers are compliant and designed for programmatic use. However, always test complex payment flows first by running simulations using `increase_simulate_inbound_ach` before deploying live code.

03 **What if I need a new account ID for my app?**

You can spin up a fresh sub-ledger bank account instantly with the `increase_create_account` tool. This gives your software dedicated, separate balances immediately.

04 **Do I have to use Sandbox mode for testing payments?**

It's highly recommended. Use the sandbox tools like `increase_simulate_inbound_ach` so you can validate payment logic and webhooks without moving any actual money.

05 **What is the difference between ACH and Wire transfers?**

ACH is generally used for standard, bulk payments to US banks. The `increase_create_wire` tool handles high-value, same-day transfers which are faster but typically incur higher fees.

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 `"increase": { "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

Increase 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 Increase. 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	Increase MCP
Server ID	`019d75b9-157d-7289-9d48-49514b06a541`
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/increase.

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