MCP SERVER | NO CODE | CLOUD HOSTED

UpCloud MCP

Manage all servers, storage, and networking via conversation.

UpCloud MCP lets your agent manage high-performance cloud infrastructure using natural language commands. From deploying and modifying servers across global zones to monitoring resource costs and handling billing summaries, you get total control of your cloud environment directly from any compatible client.

A+ Quality Score 98.33/100

cloud-servers | infrastructure-management | server-provisioning | cloud-storage | resource-monitoring | billing-management

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

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.

UpCloud MCP

46 tools available
Cloud-hosted on Vinkius

Need to juggle server status checks, storage provisioning, and networking details across dozens of browser tabs? This MCP connects your UpCloud account to any AI agent, letting you manage complex infrastructure through conversation. Instead of writing boilerplate commands or clicking through multiple dashboards, you just ask your client what you need done. Your agent acts like a dedicated cloud architect sitting next to you; it can list all available zones, create managed databases, and even write firewall rules for specific servers—all without leaving your IDE. By connecting this MCP via Vinkius, your AI gains the power to act as a comprehensive DevOps assistant right where you're already working. You handle the strategy; your agent handles the execution.

Core Capabilities

01 — Manage Server Lifecycle

You can list, start, stop, restart, and delete cloud servers across any global region.

02 — Control Storage Resources

This MCP lets you create new storage instances, modify existing ones, clone data, or restore services from backups.

03 — Configure Networking Services

You can list networks, assign IP addresses, update network details, and set up managed load balancers or private networks.

04 — Handle Account & Billing Insights

The agent pulls your account information, lists resource prices in your local currency, and delivers detailed monthly billing summaries.

05 — Build Complex Infrastructure

You can provision specialized services like managed Kubernetes clusters, databases, or object storage buckets.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and enter your specific UpCloud API credentials.

02 Connect the credential flow to any compatible AI client (like Cursor or Claude).

03 Ask your agent a natural language question, like 'What are my current billing limits?' The agent executes the necessary calls and gives you a direct answer.

The bottom line is that you bypass the command line and the web UI; your AI client handles all the API communication for you.

Built For

This MCP is built for engineers, architects, and operations staff who spend their day dealing with complex infrastructure setups. If you're tired of context switching between a terminal, a dashboard, and a billing portal just to check one thing, this is for you.

DevOps Engineer

You use the MCP to automate routine server tasks—like starting up a new test environment or updating firewall rules—without ever leaving your terminal.

Cloud Architect

You rely on this to quickly query zone availability and host details, allowing you to plan high-availability setups before writing any code.

Site Reliability Engineer (SRE)

You use it for deep resource audits, checking audit logs or retrieving billing summaries to maintain strict operational accountability.

What Changes When You Connect

01 Avoid the pain of manual monitoring. Instead of checking server status in a dashboard or running multiple shell commands to check logs, you can ask your agent for an audit log report instantly using `list_audit_logs`.

02 Save time on capacity planning. Need to know if there's room to grow? Use `list_zones` and `get_host` to query available physical hosts and zones before deploying a new application stack.

03 Maintain budget control effortlessly. Get immediate visibility into costs by calling `list_prices` or checking the full picture with `get_billing_summary`, ensuring you never overspend on cloud resources.

04 Simplify deployment workflows. Provisioning a complex setup—like a load balancer and associated network rules—becomes simple when your agent executes multiple steps using tools like `create_load_balancer` and `create_firewall_rule` in one request.

05 Speed up data management. If you need to move or secure data, you can use `backup_storage` or `restore_storage` without navigating through complex file system interfaces. Just tell your agent what needs backing up.

06 Rapidly scale resources. When a service fails or grows rapidly, you don't have to manually create everything from scratch. You simply ask the agent to deploy new components like a managed database instance using `create_database`.

Real-World Applications

The Emergency Server Check

A SRE notices an application is running slow and needs root cause analysis. They ask their agent, 'What's wrong with the web server?' The agent runs `get_server` to check its status, then calls `list_audit_logs` to see recent access patterns, providing a complete diagnosis immediately.

Planning Multi-Region Expansion

A Cloud Architect needs to build a new, highly available deployment across three regions. They ask the agent to 'List all available zones and check their capacity.' The agent uses `list_zones` and `get_host`, giving them the data needed for accurate planning.

Cost Optimization Review

A Finance Ops manager receives a vague bill. They ask their agent, 'Show me exactly how much I spent on networking this month.' The agent responds by calling `get_billing_summary` and `list_prices`, pinpointing the exact cost drivers.

Setting up Microservices

A DevOps engineer needs to deploy a new service cluster. They tell their agent, 'Set up a private network, create a load balancer, and secure it with rules.' The agent executes `create_network`, then `create_load_balancer`, followed by multiple calls to `create_firewall_rule`.

Patterns to Avoid

Over-relying on the CLI

✗ AVOID

The user manually runs `up cloud status server web-01` in the terminal, then opens a second tab to run `billing get last month`. This requires context switching and remembering multiple command flags.

✓ INSTEAD

Just ask your agent directly: 'Show me the current status of the web server and pull up the billing summary for this month.' The agent handles both actions seamlessly using tools like `get_server` and `get_billing_summary`.

Confusing storage types

✗ AVOID

The user tries to figure out if they need a standard volume or object storage, leading them to read three different vendor documentation pages.

✓ INSTEAD

Ask your agent: 'Compare managed databases and object storage capabilities.' The agent uses `list_databases` and `list_object_storages` to give you an apples-to-apples comparison.

The Right Fit

Use this MCP if managing cloud infrastructure requires deep interaction with multiple resource types—servers, networks, databases, and storage. If your job involves knowing how to provision a load balancer, check IP ranges, or audit account logs, you need this. Don't use it if you just need simple, isolated information, like reading a single price point; for that, the dedicated `list_prices` tool works fine on its own. However, if your goal is purely to manage user access and tokens without touching infrastructure details, you might only need the API token tools. This MCP shines when you need orchestration, combining multiple calls into one conversation.

The Cloud Dashboard Nightmare

Right now, managing a complex cloud setup feels like playing digital whack-a-mole. You jump from the server dashboard to check CPU usage; then you open your billing portal just to see if you hit a spending limit. If you need to change something—say, adding a new firewall rule while also cloning storage—you're juggling five different tabs and remembering half a dozen command flags.

With this MCP, all that context switching disappears. You simply tell your agent what needs doing: 'Clone the main database, increase its size, and add a firewall rule to port 80.' Your AI client handles the sequence of operations, giving you confirmation on every step without ever making you leave the chat window.

UpCloud MCP Gives You Control Over Everything

The biggest time drain is provisioning. Setting up a new service often means creating the network first, then the router, then the server itself, and finally applying firewall rules—all manual steps that invite human error.

Now, you can ask your agent to 'Provision a three-tier web application stack.' It executes `create_network`, deploys `create_router`, sets up multiple servers via `create_server`, and secures it all with `create_firewall_rule`. The whole process happens reliably in one go.

UpCloud MCP: 27 Infrastructure Tools

These tools allow your AI agent to perform nearly every infrastructure action possible in the UpCloud console, from creating databases to listing IP addresses.

#	TOOL	DESCRIPTION
01	`get_account`	Retrieves your current UpCloud account information.
02	`assign_ip`	Assigns a new IP address to an existing resource.
03	`list_audit_logs`	Lists all historical actions recorded in your account audit logs.
04	`backup_storage`	Creates a complete backup copy of a specified storage resource.
05	`get_billing_summary`	Fetches a detailed summary of your monthly billing statement.
06	`clone_storage`	Creates an exact copy of an existing storage resource.
07	`create_database`	Provisions and sets up a new managed database instance for your application.
08	`create_firewall_rule`	Creates specific rules to control network traffic entering or leaving a server.
09	`create_kubernetes_cluster`	Deploys and provisions a new Managed Kubernetes cluster (UKS).
10	`create_load_balancer`	Sets up a managed load balancer to distribute traffic across multiple servers.
11	`create_network`	Builds a new Software Defined Network (SDN) private network boundary.
12	`create_object_storage`	Creates a Managed Object Storage service for large, unstructured data files.
13	`create_router`	Provisions and configures a new network router device.
14	`create_server`	Creates a brand-new cloud compute server instance.
15	`create_storage`	Initializes and creates a new dedicated storage volume.
16	`create_api_token`	Generates a new, secure API token for programmatic access to your account.
17	`list_database_types`	Shows all available database types and the plans associated with them.

#	TOOL	DESCRIPTION
18	list_databases	Retrieves a list of managed database services currently running in your account.
19	delete_server	Delete a server
20	export_audit_logs	Export account audit logs
21	list_firewall_rules	List firewall rules for a server
22	get_server	Gets full, detailed specifications for a single server instance.
23	get_host	Get host details
24	list_hosts	Lists all physical hosts available within the cloud infrastructure zones.
25	import_storage	Transfers existing data from an external source into a managed storage volume.
26	list_ips	Retrieves a list of all IP addresses associated with your account.
27	get_kubernetes_kubeconfig	Downloads the necessary credentials (kubeconfig) to connect to your managed cluster.
28	list_kubernetes_clusters	Lists all Managed Kubernetes clusters currently deployed in your account.
29	list_load_balancers	Retrieves a list of active managed load balancer services.
30	list_networks	Lists all Software Defined Private Networks you have set up.
31	list_object_storages	Retrieves a list of managed object storage services.
32	list_prices	Fetches the current cost price for various cloud resources in your account's currency.
33	release_ip	Release an IP address
34	restart_server	Restart a server
35	restore_storage	Restore storage from backup
36	list_routers	List routers
37	list_servers	List all servers
38	start_server	Start a server
39	stop_server	Stop a server

#	TOOL	DESCRIPTION
40	`list_storages`	List all accessible storages
41	`list_api_tokens`	List API tokens
42	`update_ip`	Modify PTR record or floating IP attachment
43	`update_network`	Modify network details
44	`update_server`	Modify server configuration
45	`update_storage`	Modify or resize storage
46	`list_zones`	List available zones

See It in Action

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

U Show my UpCloud account information and credits.

I've retrieved your account details. You currently have $150.00 in credits remaining, and your resource limits allow for up to 20 concurrent servers. Would you like to see your billing summary for this month?

U List all my active servers on UpCloud.

Fetching your server list... I found 3 servers: 'web-prod-01' (Running), 'db-master' (Running), and 'test-bench' (Stopped). Which one would you like to manage?

U Restart the server with UUID 00264663-890e-473c-9189-9e87483660c2.

Initiating restart for server 00264663-890e-473c-9189-9e87483660c2... The command has been sent successfully. The server should be back online in a few moments.

Frequently Asked Questions

01 **How does the UpCloud MCP handle billing summaries?**

The agent retrieves a detailed monthly summary using the `get_billing_summary` tool. This gives you an accurate breakdown of your resource usage and costs for the period.

02 **Can I use the UpCloud MCP to manage my servers?**

Yes, you can start, stop, restart, or delete any server instance using tools like `start_server`, `stop_server`, and `delete_server`. This gives full lifecycle control.

03 **What if I need to change a network setting?**

You use the `update_network` tool. You just describe the modification you want, and the agent applies it correctly across your SDN private networks.

04 **Does UpCloud MCP help with IP addresses?**

Absolutely. The agent lets you list all IPs using `list_ips`, assign a new one with `assign_ip`, or release an old address via `release_ip`.

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 `"upcloud": { "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

UpCloud 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 UpCloud. 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	UpCloud MCP
Server ID	`019e3902-d4fd-73b7-bd66-28c48298d13e`
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/upcloud.