MCP SERVER | NO CODE | CLOUD HOSTED

JD Cloud / 京东云 MCP

Manage VMs, storage, and billing via conversation.

JD Cloud / 京东云 connects your AI client directly to China's leading cloud infrastructure platform. It lets you manage complex resources—from virtual machines and network configurations to object storage buckets and detailed billing reports—all through natural conversation. Stop navigating confusing consoles; get accurate, real-time operational data instantly.

A+ Quality Score 100/100

cloud-computing | vm-management | storage-buckets | billing-analysis | infrastructure-as-code | enterprise-cloud

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/jd-cloud

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.

JD Cloud / 京东云 MCP

8 tools available
Cloud-hosted on Vinkius

Forget logging into the JD Cloud Console just to check a few metrics. This MCP turns your cloud infrastructure management into a simple chat with your AI agent. Whether you're auditing spending or checking if an application is running correctly, you just ask. Your agent handles the complexity of querying VM status, listing all storage buckets across regions, and pulling comprehensive cost reports for specific time frames. It gives you precise answers about everything from VPC networks to IAM account details—all without needing a single click through complex menus. By using this MCP via Vinkius, your AI client becomes an immediate cloud operations co-pilot that provides accurate data straight into your conversation flow.

Core Capabilities

01 — Monitor Compute Status

You can list all virtual machines and retrieve their current metadata to check operational status.

02 — Audit Cloud Storage

The MCP lets you see all existing Object Storage Service buckets and manage cloud disk resources across different regions.

03 — Review Network Topology

It retrieves details about Virtual Private Clouds and helps audit your overall network setup.

04 — Analyze Spending

You get accurate billing summaries for specific time ranges, keeping track of cloud costs.

05 — Check Account Health

The agent can verify project connectivity, active regions, and retrieve detailed IAM account profile information.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your JD Cloud Access Key, Secret Key, and Region details.

02 Connect the service credentials to your preferred AI client (Claude, Cursor, etc.).

03 Ask your agent a question—like 'What were my costs last month?' or 'List all active VMs.'—and receive structured data in response.

The bottom line is that you manage complex cloud assets using simple commands, without ever seeing the underlying console UI.

Built For

This MCP is built for Ops Engineers and IT Finance managers who are tired of jumping between dashboards. If your job involves auditing infrastructure health or tracking monthly burn rates across multiple cloud services, this tool saves hours.

Site Reliability Engineer (SRE)

Runs automated checks on system health, listing virtual machines and checking network details to preemptively catch outages.

DevOps Engineer

Integrates infrastructure auditing into daily routines, using the MCP to check connectivity or list cloud disks before a deployment run.

IT Financial Analyst

Runs monthly audits by requesting billing summaries and cross-referencing those costs against specific resource usage like OSS buckets.

What Changes When You Connect

01 Avoids console navigation. Instead of clicking through multiple dashboards to audit resource usage, you simply ask your agent for the details using tools like list_vm_instances or list_oss_buckets.

02 Improves cost control. You can generate a precise financial picture by requesting billing summaries with get_billing_summary, instantly pinpointing where every penny went last quarter.

03 Saves setup time on audits. Need to check the network? Use list_vpc_networks to see all your private cloud setups without manually checking each subnet GUI.

04 Consolidates account checks. You can verify project connectivity and view IAM details with get_account_profile in one single query, rather than running five separate reports.

05 Handles complex resource types. It gives you visibility into everything from attached disks (list_cloud_disks) to entire CI/CD pipelines (list_cicd_pipelines).

Real-World Applications

Quarterly Cost Review

The finance team needs to prove cost accountability for Q3. They ask their agent, 'What were the total costs from VMs and OSS storage in July?' The agent uses get_billing_summary and list_oss_buckets to deliver a single, actionable report.

Debugging an Application Failure

A developer reports that the web app is slow. They ask their AI client, 'List all running VMs in our production VPC.' The agent uses list_vm_instances and list_vpc_networks to immediately give them a list of candidates for investigation.

New Project Onboarding

A new team needs to set up storage. Instead of reading documentation, they ask the agent to 'Show me all existing object storage buckets.' The agent uses list_oss_buckets and can even check if a suitable disk space is available via list_cloud_disks.

Pre-Audit Checklist

The SRE needs to confirm the health of all components before a major migration. They query the agent for 'account profile' details, check network topology with list_vpc_networks, and ensure no old pipelines are running via list_cicd_pipelines.

Patterns to Avoid

Manual Dashboard Jumping

✗ AVOID

Trying to compare billing data against actual VM usage by logging into the Billing Console, then switching to the Compute Console, and manually cross-referencing timestamps.

✓ INSTEAD

Use get_billing_summary alongside list_vm_instances. Your agent pulls both datasets together in a single conversation flow, giving you immediate comparison.

Forgetting Resource Scope

✗ AVOID

Assuming all storage buckets are visible simply because they exist within the same project folder.

✓ INSTEAD

Always run list_oss_buckets. This command guarantees visibility across all configured object storage services, no matter where they live.

Ignoring Account Context

✗ AVOID

Running a query about resources without knowing which region or project the assets belong to.

✓ INSTEAD

Always start by asking get_account_profile. This verifies your scope and confirms you're querying the correct organizational unit before running other resource-specific tools.

The Right Fit

Use this MCP if your primary job involves continuous auditing, cost management, or troubleshooting across multiple distinct cloud services (compute, storage, network). You need an agent that can act as a unified dashboard. Don't use it if you only ever deal with one isolated service—for example, if you *only* manage email sending, you don't need this. If your task is highly specialized and confined to a single API function (like just managing user accounts), a more focused tool might be better. However, because this MCP combines tools for list_vm_instances, get_billing_summary, and list_oss_buckets into one conversational interface, it's the default choice for cloud architects and finance teams.

Cloud auditing is a painful click-through process.

Right now, checking your cloud spending requires juggling three different interfaces: you start in the Billing Console to see total costs. Then you jump to the Compute section to manually check if resource usage matches what was billed. If you need to audit storage buckets, you open a third tab dedicated only to OSS.

With this MCP, you just talk to your agent. You ask about billing and VM status in one prompt. The agent uses its tools—like get_billing_summary and list_vm_instances—to pull the data from all those separate systems and give you one coherent answer.

JD Cloud / 京东云 MCP: Get a complete resource picture.

You no longer have to copy-paste IDs or remember which tab holds the network topology details. The agent handles that complexity, using list_vpc_networks and get_vm_detail to build you a full map of your infrastructure's physical boundaries.

The difference is control. You move from being a manual data aggregator to giving natural language commands that execute complex operations across your entire cloud setup.

JD Cloud / 京东云: 8 Tools for Cloud Management

Use these eight tools to perform specific cloud operations, including listing resources, auditing finances, retrieving VM details, and managing network setups.

#	TOOL	DESCRIPTION
01	`get_account_profile`	Retrieves general information about the connected IAM user account.
02	`get_billing_summary`	Pulls a detailed overview of cloud spending for a specified time period.
03	`get_vm_detail`	Fetches the specific metadata and details for an individual virtual machine instance.
04	`list_cicd_pipelines`	Lists existing Continuous Integration/Continuous Deployment pipelines within the account.
05	`list_oss_buckets`	Lists all configured Object Storage Service buckets in the account.
06	`list_vm_instances`	Provides a list of all virtual machines currently managed by the cloud account.
07	`list_vpc_networks`	Lists and provides details for all Virtual Private Clouds (VPC) networks.
08	`list_cloud_disks`	Retrieves a comprehensive list of attached or unattached block storage disks.

See It in Action

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

U List all my running virtual machines in region 'cn-north-1'.

I've retrieved your VM instances for the Beijing region. You have 3 active instances: 'Web_Server_01' (Running), 'DB_Master' (Running), and 'Test_Lab' (Stopped). Would you like to see the detailed specs for the 'Web_Server_01'?

U Check our JD Cloud billing summary from October 1st to October 15th.

I've retrieved the billing summary for your account. For the requested period, the total expenditure was 1,240.50 CNY. The largest cost contributors were VM instances (65%) and OSS storage (20%). Should I check the detailed usage for VM resources?

U Show me the list of Object Storage (OSS) buckets in my account.

I've retrieved your OSS bucket list. You have 5 buckets configured, including 'assets-prod', 'backup-logs', and 'user-uploads'. Would you like me to check the region and creation date for a specific bucket?

Frequently Asked Questions

01 **Does JD Cloud / 京东云 MCP handle billing for non-CNY currencies?**

It retrieves comprehensive billing summaries, showing costs in the currency designated by your account. You just need to confirm the correct region access key when setting up.

02 **What if I have hundreds of VMs? Can JD Cloud / 京东云 MCP list them all?**

Yes. The list_vm_instances tool gathers a full inventory of your virtual machines, allowing you to then use get_vm_detail on any specific instance for deeper analysis.

03 **Can I check my network configuration with JD Cloud / 京东云 MCP?**

Absolutely. The list_vpc_networks tool gathers all details about your Virtual Private Clouds, helping you audit the boundaries and connectivity of your services.

04 **Is this good for checking storage buckets? Does JD Cloud / 京东云 MCP support it?**

Yes. The list_oss_buckets tool is specifically designed to inventory every Object Storage Service bucket, ensuring you know exactly what data you're storing and where.

05 **How does get_vm_detail work if I don't know the VM ID?**

You first run list_vm_instances to get a list of available IDs. Then, you pass one of those specific IDs to the get_vm_detail tool for full metadata.

06 **How do I find my JD Cloud Access Key and Secret Key?**

Log in to the [JD Cloud Console](https://console.jdcloud.com/), navigate to [Account Management] → [Access Key Management] to find or create your unique credentials.

07 **What regions are supported?**

JD Cloud supports multiple regions including `cn-north-1` (Beijing), `cn-east-1` (Suzhou), `cn-east-2` (Shanghai), and `cn-south-1` (Guangzhou). Ensure you specify the correct region where your resources are located.

08 **Does this server handle JDCLOUD2-HMAC-SHA256 signatures?**

Yes! The server automatically calculates the required Signature v2 (JDCLOUD2-HMAC-SHA256) for every request using your provided Secret Key, ensuring high security for your cloud orchestration.

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 `"jd-cloud": { "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

JD Cloud / 京东云 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 JD Cloud / 京东云. 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	JD Cloud / 京东云 MCP
Server ID	`019d844a-7b49-702a-82cd-26782d7492f5`
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/jd-cloud.