MCP SERVER | NO CODE | CLOUD HOSTED

JumpCloud MCP

Audit access control and manage user identities.

JumpCloud MCP connects your AI client directly to an enterprise-grade directory service for managing users and systems. Your agent can check account details, audit group memberships, view all connected applications, or list managed hardware across your organization.

A+ Quality Score 100/100

directory-services | sso | user-management | it-administration | access-control

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

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.

JumpCloud MCP

10 tools available
Cloud-hosted on Vinkius

Need to know who has access to what? This MCP gives your AI client the full picture of your company's digital identity landscape. It lets you query user records and system groups without logging into a dozen different dashboards. Your agent can check account metadata, track which applications are connected via SSO, or list every managed device in your network. JumpCloud handles everything from directory configurations to security policies, providing all that data through one open standard connection. Whether you're running compliance checks or just onboarding a new team, this MCP lets you automate IT administration tasks right where you work. By connecting this MCP through Vinkius, you give any compatible AI client direct access to core identity services.

Core Capabilities

01 — Check User Identity Details

Retrieve specific account metadata, group memberships, and security settings for individual users.

02 — Audit System Hardware Status

List all corporate systems managed by JumpCloud to audit hardware inventory and device compliance.

03 — Map Access Policies

View defined security policies, like disk encryption requirements or firewall rules, across the entire fleet.

04 — Verify Organizational Structure

List all user groups and system groups to map out your organizational access control model.

05 — Review Connected Services

Audit which SaaS applications are integrated into the directory via Single Sign-On (SSO).

One Click on Vinkius — From Prompt to Execution

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

01 Your agent uses your AI client to authorize access to JumpCloud through Vinkius.

02 The agent calls a specific tool, such as `list_users` or `list_policies`, passing necessary parameters (e.g., 'all' users, 'disk encryption' policy).

03 JumpCloud executes the request and sends back structured data—like user IDs, group lists, or application names—which your agent uses to generate a final report.

The bottom line is you get a single API endpoint that lets your AI client read and audit identity information across your entire infrastructure.

Built For

This MCP is for the SecOps engineer who needs to prove compliance quickly, or the IT administrator tired of manually checking user permissions. If your job involves auditing access control or managing large fleets of devices, this tool saves hours of clicking through disparate consoles.

Security Engineer

They use it to audit system security policies and check group memberships to ensure that only authorized personnel have access to sensitive resources.

IT Operations Manager

They rely on this MCP to track managed systems, list user accounts, and confirm the status of directory configurations across all departments.

Compliance Auditor

They use it to generate reports listing configured directories and applications, proving that access control policies are uniformly applied company-wide.

What Changes When You Connect

01 Instead of clicking through multiple portals, your agent can instantly run `list_users` to get a complete roster of accounts for auditing purposes.

02 You gain immediate visibility into compliance status. Running the `list_policies` tool shows every security rule defined on your fleet, making audits simple.

03 The MCP helps you track hardware and device compliance by running `list_systems`, giving you an accurate inventory without manual checks.

04 Mapping access is faster than ever. You can use `list_user_groups` combined with `list_system_groups` to understand exactly who belongs where.

05 It streamlines auditing connected services. Using `list_applications` quickly shows which third-party SaaS tools require SSO credentials.

Real-World Applications

Investigating unauthorized access post-offboarding

The HR team asks, 'Who still has access to the main network?' Your agent uses `list_users` and then calls `get_user` for specific accounts. It reports on group memberships and security settings, allowing you to confirm exactly which credentials need disabling.

Preparing for a PCI compliance audit

A consultant needs proof that all sensitive data endpoints are encrypted. Your agent runs `list_policies` to retrieve details on mandatory disk encryption and then uses `list_systems` to confirm which managed devices adhere to the rule.

Mapping a new department's permissions

A manager needs to know what access rights are assigned to their new team. Your agent first runs `list_user_groups` and then uses this information alongside `list_directories` to show the organizational structure and its linked identity sources.

Auditing network entry points

The security team suspects a weakness in remote access. Your agent runs `list_networks` to see all RADIUS authentication settings, then uses `list_applications` to check which services rely on SSO for connection.

Patterns to Avoid

Trying to manage users via ticketing system

✗ AVOID

An agent finds a user account needs deactivation and drafts a ticket. The process stalls because the service desk agent has to manually log into JumpCloud, find the ID, and click 'Deactivate'.

✓ INSTEAD

Instead of creating tickets, use your MCP. Your agent can call `get_user` to confirm credentials and then execute the necessary action directly against the directory via the AI client.

Only auditing one type of access

✗ AVOID

The team only checks user groups but misses that a policy change requires updating device-specific settings. They assume group membership is enough.

✓ INSTEAD

Don't stop at `list_user_groups`. Always check compliance by calling `list_policies` and cross-reference the results with `list_system_groups` to ensure policies apply correctly.

Ignoring device inventory gaps

✗ AVOID

The security team runs an access audit but doesn't know which physical devices haven't been provisioned or are running old OS versions.

✓ INSTEAD

Before auditing access, run `list_systems` first. This gives you the full list of managed hardware and their current OS status, letting you focus your policy checks.

The Right Fit

Use this MCP if your primary job is directory management, identity auditing, or compliance reporting across a large technical environment. You need to know *who* has access to *what*, and whether that setup follows defined security policies. This tool excels when you must correlate user identities with system health (e.g., 'Does User X on Device Y have Policy Z applied?').

Don't use this if your pain point is purely ticketing or incident response; you need a dedicated ITSM platform for that. Also, if your problem involves complex data transformation or business logic outside of identity management, you might be better off using a general-purpose workflow automation tool instead of relying solely on `list_users` and group lookups.

Tracking down who has access to what used to take half a day.

Right now, checking user permissions means logging into the directory console, pulling reports of users, then switching over to the device management portal. You have to cross-reference group names against policy lists and manually check if every system is reporting its compliance status—it's a mess of clicks and copy-pasting.

With this MCP, your agent handles the complexity. It pulls data from identity sources like LDAP or AD directly into one feed. You just ask: 'Who can access the financial server?' And you get an instant report combining user IDs, group membership, and system compliance status.

JumpCloud MCP Gives You Real-Time Access Visibility

The manual steps that vanish are the need to switch between identity consoles, cross-reference `list_user_groups` with `list_system_groups`, and then validate against security policies found in `list_policies`. All of this is consolidated into a single data flow.

Now, instead of spending hours correlating data points across multiple dashboards, you ask your agent one question and get the definitive answer. It's that simple.

JumpCloud: 10 Tools for Directory Management

These tools let your agent check everything from individual user details and group structures to overall security policies across your connected network.

#	TOOL	DESCRIPTION
01	`get_user`	Retrieves detailed account metadata, group memberships, and security settings for a specific user.
02	`list_applications`	Lists all configured SSO applications used to control software access.
03	`list_commands`	Shows saved management commands that can be run for automation auditing.
04	`list_directories`	Lists all configured identity sources, such as LDAP or Google directories.
05	`list_networks`	Shows details about all RADIUS networks used for WiFi and VPN authentication.
06	`list_policies`	Lists current system security policies, such as disk encryption or firewall rules.
07	`list_system_groups`	Shows predefined organizational groupings for devices, like 'Employee Laptops'.
08	`list_systems`	Returns hostnames and IDs of all company hardware managed by JumpCloud.
09	`list_user_groups`	Lists the defined user groups, helping map out organizational access control structures.
10	`list_users`	Provides a list of all users in JumpCloud, acting as the primary point for identity auditing.

See It in Action

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

U List all users in my JumpCloud directory.

I'll fetch the complete list of users from your JumpCloud account.

U Show me the managed systems currently active.

I'll retrieve the list of systems managed by JumpCloud for you.

U Check the user groups in my organization.

I'll look up the list of configured user groups in your account.

Frequently Asked Questions

01 **How does JumpCloud MCP help with user deactivation?**

You use `get_user` to retrieve full account metadata, confirming current group memberships and security settings before initiating any changes. This ensures you deactivate the right access points.

02 **Can I audit all my connected SaaS apps with JumpCloud MCP?**

Yes, calling `list_applications` provides a comprehensive inventory of every Single Sign-On (SSO) application integrated into your directory. This is crucial for security audits.

03 **What if I need to check device compliance? Use JumpCloud MCP.**

Run `list_systems` to get a list of all managed hardware IDs and hostnames. You can then cross-reference this with `list_policies` to confirm which systems meet required security standards.

04 **Does JumpCloud MCP handle directory mapping?**

The tool is built for it. By using `list_directories`, you can see all configured identity sources, whether they are LDAP, AD, or Google-based.

05 **Which tools list user accounts in JumpCloud MCP?**

`list_users` provides the primary roster of users. For deeper checks on a single person, use `get_user` to see their specific group memberships and security settings.

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 `"jumpcloud": { "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

JumpCloud 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 JumpCloud. 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	JumpCloud MCP
Server ID	`019d75be-62ed-73bb-999e-ce3798b669d5`
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/jumpcloud.