MCP SERVER | NO CODE | CLOUD HOSTED

Jamf Pro MCP

Audit your entire Mac and mobile fleet status.

Jamf Pro MCP connects your AI agent directly to the Jamf Pro API, letting you audit and manage entire Apple device fleets. Use it to list all managed computers, check mobile security status, track inventory details like disk encryption, or see which users are assigned where. It's essential for IT teams needing a full view of their hardware and software deployment.

A+ Quality Score 100/100

mobile-device-management | it-automation | asset-tracking | endpoint-security

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

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.

Jamf Pro MCP

10 tools available
Cloud-hosted on Vinkius

This MCP lets your AI client work with the Jamf Pro API to manage your entire Apple ecosystem. Your agent can list all managed computers and mobile devices, giving you an immediate inventory count and status overview. Need to know who's using what? You can look up users and see which departments they belong to. The system also tracks physical locations by listing buildings or analyzing device distribution across specific categories. For automation tasks, your AI client doesn't just read data; it pulls details about available software packages or existing management scripts for remote execution. When you connect this via Vinkius, you give your agent the full capability to audit everything from user accounts and computer serial numbers to department assignments in one place.

Core Capabilities

01 — Audit all managed computers

The MCP retrieves detailed records for every Mac on the network, including installed applications and disk encryption status.

02 — Investigate specific mobile devices

You can pull deep data on any single mobile asset, showing its operating system version, assigned user, and security state.

03 — Map organizational structure

The MCP lists all configured departments, buildings, and management categories to understand how devices are distributed across the company.

04 — Track users and assets

You can pull a list of every user in the system and view which specific machines or mobile devices they are associated with.

05 — Review deployment resources

The MCP lists all available software packages, management scripts, and custom categories ready for deployment across the fleet.

One Click on Vinkius — From Prompt to Execution

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

01 You tell your AI client exactly what you need to audit—for example, 'Find all Macs without disk encryption.'

02 The client calls the relevant tool in this MCP (like `list_computers` or `get_computer`), which executes a request against the Jamf Pro API.

03 Your agent receives structured data containing the specific inventory details, user assignments, or deployment status you requested.

The bottom line is that your AI client uses this MCP to talk directly to Jamf Pro, giving it real-time asset and user information.

Built For

This is for the IT Operations team or Endpoint Security Engineers who spend their days manually cross-referencing spreadsheets with dashboard reports. If you're tired of clicking through multiple Jamf Pro sections just to figure out if a device meets compliance standards, this MCP gets your agent working.

IT Operations Engineer

Audits the entire Mac fleet using `list_computers` and checks individual machine details with `get_computer`, ensuring all assets are accounted for.

Security Analyst

Uses `get_mobile_device` to verify security status and OS versions on specific mobile assets, feeding data into compliance reports.

System Administrator

Maps out the organizational structure by listing buildings or departments (`list_buildings`, `list_departments`) before rolling out new software packages via `list_packages`.

What Changes When You Connect

01 Stop manually checking compliance. By using `get_computer`, you can instantly audit a machine's disk encryption status, which is crucial for security reporting.

02 Get location context immediately. Tools like `list_buildings` or `list_departments` let your agent map device distribution across physical sites and business units.

03 Improve asset visibility by using `list_mobile_devices`. You can quickly get a list of all managed phones and tablets, speeding up inventory checks.

04 Streamline software deployments. Instead of guessing what's available, use `list_packages` to see every software package ready for rollout across the entire organization.

05 Simplify user-asset mapping. Running `list_users` lets your agent instantly pair users with their assigned devices, making audit reports simple and accurate.

Real-World Applications

The Quarterly Compliance Audit

A security analyst needs to prove all corporate Macs have disk encryption enabled. They ask the agent to run `list_computers` first, then iterate through the results calling `get_computer` for each one, compiling a single report showing every machine's current status.

Onboarding a New Department

A sysadmin needs to provision 50 new employee laptops. They use `list_departments` to find the correct unit ID and then call `get_computer` repeatedly to ensure every device assigned to that department is accounted for.

Investigating Lost Devices

A manager loses track of company phones. They ask their agent to run `list_mobile_devices` to get a master list, and then use `get_mobile_device` on specific IDs to verify the last known OS version.

Pre-Rollout Check

An IT team is prepping an OS update. Before deployment, they check `list_packages` to confirm the correct software is available and then use `list_scripts` to ensure any necessary pre-run scripts are ready.

Patterns to Avoid

Assuming a full device list exists.

✗ AVOID

The user just asks, 'What devices do we have?' without specifying the type or scope, leading to vague partial results.

✓ INSTEAD

Be specific. Always start by calling `list_computers` if you need Macs, or use `list_mobile_devices` if you only want phones and tablets.

Overlooking physical location data.

✗ AVOID

Reporting device counts without knowing which building they are in, making the report useless for facility planning.

✓ INSTEAD

Before reporting totals, run `list_buildings` to establish the scope. Then, use tools like `get_computer` while filtering by building ID.

Mixing up user data with hardware status.

✗ AVOID

Listing users and then separately listing computers without connecting them, resulting in two unrelated lists that need manual comparison.

✓ INSTEAD

Use `list_users` to get the roster, then call `get_computer` using the user ID or device name to correlate who has which machine.

The Right Fit

Use this MCP if your primary job is auditing hardware inventory and maintaining compliance across a mixed fleet of Apple computers and mobile devices. You need deep-dive data points, like disk encryption status (`get_computer`) or OS versions (`get_mobile_device`), which only Jamf Pro holds.

Don't use this MCP if your goal is simply to send out mass announcements or manage ticketing workflow—you'd be better off using a dedicated messaging tool. Also, don't use it if you just need generic user directories without asset linkage; `list_users` gives names but not device status.

If you only need basic reporting (e.g., 'How many Macs do we have?'), this is perfect. But if you need to know *why* a Mac has that specific OS version, or which building it's assigned to, you need the deep tool set provided here.

The Manual Chore of Auditing Device Compliance

Right now, checking compliance means jumping between dashboard views and spreadsheets. You pull a list of 500 employee Macs, then you have to manually open the detailed view for each one just to confirm if disk encryption is active or what OS version it's running. It's tedious copy-pasting, cross-referencing IDs, and spending hours chasing missing data points.

With this MCP, your agent handles that entire process in a single sequence of calls. You ask for compliance status across the fleet, and your agent automatically uses tools like `list_computers` followed by `get_computer` on every asset to build you one complete, accurate report.

Get Full Visibility with Jamf Pro's Tools

The biggest time sink is correlating who has what. You pull the user list from one place and the device inventory from another, forcing you to

Instead, your agent combines `list_users` with the asset data. It gives you immediate, correlated knowledge: User X is assigned Device Y, which

manually match names and IDs in a separate spreadsheet. This step alone can take hours.

belongs to Department Z. The whole picture just appears.

Jamf Pro with 10 Tools

Use these tools to run specific queries against the Jamf Pro API, allowing your AI client to retrieve detailed asset records and system configurations.

#	TOOL	DESCRIPTION
01	`get_computer`	Retrieves detailed information for one specific computer, including its installed apps and disk encryption status.
02	`get_mobile_device`	Pulls comprehensive data on a single mobile device, showing its security status, assigned user, and OS version.
03	`list_buildings`	Lists all physical buildings configured within the Jamf Pro management system.
04	`list_categories`	Retrieves a list of all management categories, useful for understanding the hierarchy used in device grouping.
05	`list_computers`	Lists every managed computer on the network, providing names, IDs, and serial numbers for fleet auditing.
06	`list_departments`	Provides a list of all configured business departments, allowing analysis of device distribution by unit.
07	`list_mobile_devices`	Lists every managed mobile device, providing their names, IDs, and models for asset auditing.
08	`list_packages`	Returns a list of all software packages available in Jamf Pro's distribution points for deployment checks.
09	`list_scripts`	Lists all management scripts stored within Jamf Pro, which can be used for remote automation execution.
10	`list_users`	Retrieves a full list of users in the system and identifies their current associations with devices.

See It in Action

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

U List all computers managed in Jamf Pro.

I'll fetch the list of managed computers and their inventory details for you.

U Show me details for mobile device ID '456'.

I'll retrieve the full inventory and management data for that mobile device.

U List all management scripts configured in the system.

I'll look up the list of available management scripts in Jamf Pro.

Frequently Asked Questions

01 **Can Jamf Pro MCP list all managed hardware?**

Yes, the MCP can list both Mac computers using `list_computers` and mobile devices via `list_mobile_devices`, giving you a comprehensive inventory count.

02 **How do I check if an individual computer is encrypted?**

You call the `get_computer` tool, passing in the specific device ID. The returned data includes the current disk encryption status for that machine.

03 **Does Jamf Pro MCP help me find devices by location?**

Yes, you can audit distribution using tools like `list_buildings` and `list_departments`, allowing you to filter or analyze assets based on their assigned physical site or business unit.

04 **What if I need to see available software packages?**

Use the `list_packages` tool. It retrieves a list of every `.pkg` or `.dmg` file available in Jamf Pro's distribution points, so you know exactly what can be deployed.

05 **Can I see which user owns a specific device?**

You use `list_users` to view the full roster and then cross-reference that data with device details retrieved from tools like `get_computer` or `get_mobile_device`.

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 `"jamf-pro": { "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

Jamf Pro 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 Jamf Pro. 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	Jamf Pro MCP
Server ID	`019d75bc-4778-722c-b9bb-6186a8738eea`
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/jamf-pro.