

MCP SERVER

NO CODE

CLOUD HOSTED

# Ping Identity MCP

Audit every user identity and security policy in one conversation.

Ping Identity MCP lets you manage complex enterprise identities and access rules directly through your AI client. You can audit users, check security policies, and review application connections across PingOne without navigating multiple admin consoles. It gives you full control over user lifecycle management, group memberships, and zero-trust risk settings.

**A+** Quality Score 100/100

sso

mfa

zero-trust

iam

user-provisioning

authentication



# 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.

### 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.

### 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.

### 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**

Encryption at rest

**Ed25519**

PKI vault signatures

**24h TTL**

Ephemeral session keys

**V8 Isolate**

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.

### 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.

### 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.

### 06 — Honeypot Trap System

Phantom credentials are injected into isolated environments. If a honeypot 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.

**FinOps**

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

**Firewall & DLP**

PII redaction activity, sensitive data protection counters, and security event timeline.

**Agent Activity**

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

**Tool Health**

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

**Incident Log**

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

Get started at [cloud.vinkius.com](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Ping Identity MCP

10 tools available

Cloud-hosted on Vinkius

Managing who gets into the network and what they can do is a headache, especially when policy changes happen fast. This MCP connects your AI client to your entire Ping Identity environment. You stop switching between consoles and start talking to it instead. Need to check if a user still exists? Just ask. Want to see which apps are connected or audit who belongs to a specific group? Your agent handles the requests, pulling detailed data on everything from application grants to complex security policies.

It's all about keeping your identity posture clean and secure. Whether you need to list every internal user profile, review active risk management rules that trigger MFA, or delete an account completely, this MCP gives your AI client the necessary tools. Because Vinkius hosts this connection, you get access to the whole suite of IAM functions from one place.

---

## Core Capabilities

### 01 — Audit User Status

Get complete details for any user profile, including their current metadata and status in the system.

### 03 — Inspect Security Rules

View active risk management policies that dictate real-time authentication requirements, like mandatory MFA or location checks.

### 05 — View Group Membership

Retrieve detailed information on specific groups used for assigning permissions across the enterprise.

### 02 — Review Applications

List and audit all federated web or native applications connected to your PingOne environment.

### 04 — Manage User Accounts

List all users in the directory and execute a hard deletion of an identity, purging all associated credentials.

### 06 — Examine Sign-On Flows

Check logical sign-on policies to see what conditions, like required passwords or biometrics, must pass before access is granted.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/ping-identity](https://vinkius.com/mcp/ping-identity) — connect your AI agent in three steps.

- 01 Subscribe to this MCP and provide your PingOne Environment ID along with an API token.
- 02 Your AI client connects through the Vinkius platform, authenticating against your credentials.
- 03 You simply instruct your agent in natural language—for example, 'Show me all users who haven't logged in for 90 days.' — and the MCP executes the necessary actions.

The bottom line is, you control your identity infrastructure by talking to it instead of clicking through menus.

---

## Built For

This connector is for IAM Administrators or Security Engineers who are sick of context switching between the PingOne dashboard and a dozen other compliance tools. If your job involves auditing user access, checking policy gaps, or running off-boarding procedures, this saves hours.

### IAM Administrator

Runs daily audits to verify application settings, manages user groups for role mapping, and ensures user lifecycles are handled correctly.

### Security Engineer

Reviews real-time risk policies and sign-on flows instantly. They check for impossible travel rules or anonymous network access before a breach happens.

### IT Support Specialist

Needs to quickly pull specific user metadata, verify credential health, or confirm if an account can be safely deleted without breaking linked services.

---

## What Changes When You Connect

- 01 Audit entire application footprints instantly. Instead of manually navigating multiple tabs to check which Web or SPA apps are federated, you use `list_applications` and get a comprehensive list immediately.

- 
- 02 Enforce Zero Trust compliance easily. You can review real-time risk policies using `list_risk_policies`, seeing exactly which rules dictate MFA prompts or block impossible travel attempts without leaving your chat window.

---

  - 03 Handle user offboarding with precision. Use the `delete_user` tool to hard delete an account, ensuring all associated sessions, app scopes, and credentials are purged safely in one API call.

---

  - 04 Understand complex access structures quickly. To map permissions, check group roles using `list_groups` or review specific users' full context by calling `get_user`, eliminating guesswork about who has what access.

---

  - 05 Verify policy complexity without the console headache. You can view sign-on flows with `list_sign_on_policies` to confirm if complex rules, like mandatory biometrics, are active before a user gets access.
- 

---

## Real-World Applications

### Investigating an Anomalous Login

A security team member noticed unusual login activity. They ask their agent to check the 'Impossible Travel' policies using `list_risk_policies`. The agent responds by detailing that a rule is active, requiring MFA whenever a user jumps across continents in an hour, immediately flagging potential compromised accounts.

### Removing a Former Employee

An IT support specialist needs to offboard an employee immediately. They use the `delete_user` tool with the user's ID. The agent confirms that not only is the identity removed, but all associated MFA linkages and secondary sessions are also purged from the directory.

### Auditing Contractor Access

The compliance officer needs to know which external parties are connected. They ask their agent to `list_populations`. The tool replies with a breakdown of 'Contractors' and 'Partners', confirming that these isolated populations have different password expiration rules than full-time employees.

### Reviewing App Security Gaps

A developer needs to check an old application. They ask their agent to `get_application` for a specific federated app. The tool returns detailed configuration, showing if the app is using secure PKCE flows or if it's still relying on less secure methods.

---

# Patterns to Avoid

---

## Assuming simple user lists are enough

### X AVOID

Manually asking to list users and assuming you know what permissions they have. You get a basic list, but that doesn't tell you if their account is active or what groups they belong to.

### ✓ INSTEAD

First, run ``list_users`` to see the full directory roster. Then, for critical profiles, use ``get_user`` and follow up with ``list_groups`` to map out exactly what access rights are assigned.

---

## Fixing policies by guessing

### X AVOID

Seeing a login failure and trying to adjust the sign-on rules without knowing which policy is active. This can accidentally lock people out or leave security gaps open.

### ✓ INSTEAD

Always start by running ``list_sign_on_policies`` to see all logical flows, then use ``list_risk_policies`` to check what real-time conditions (like location) are causing the failure before making any changes.

---

## Skipping population checks

### X AVOID

Treating contractors and employees the same way. You might accidentally apply full employee rules to a B2B client, violating compliance mandates.

### ✓ INSTEAD

Always call ``list_populations`` first. This shows you distinct groups like 'Contractors' or 'Partners', ensuring you manage each segment with its unique set of security parameters.

---

## The Right Fit

Use this MCP if your core job involves auditing, verifying, and managing the technical relationship between identities and permissions within Ping Identity. This is for deep IAM work—you need to confirm policy compliance, track down who owns an application grant, or perform clean offboarding.

Don't use it if you are only looking for basic information, like a simple contact directory search (use your local LDAP tools instead). Also, if your goal is just to write code that uses user IDs without checking their current status or policy constraints, this MCP might be overkill. You need the granular control of specific actions: Use `delete_user` when you are certain an account must vanish; use `get_group` when mapping roles; and always check `list_risk_policies` before making any change to authentication.

---

## The Headache of Identity Auditing

Today, checking a user's full security profile is a multi-step ordeal. You jump into the admin console, find the user ID, then click tabs for groups, applications, and policies. If you need to know if an application connection is secure or which population they belong to, you're clicking through five different menus, copying IDs, and pasting them into spreadsheets just to prove compliance.

With this MCP, that manual chore disappears. You simply ask your agent: 'Audit the access footprint for user X.' The system runs checks across group memberships, federated apps, and risk policies in seconds, giving you a clean, consolidated report telling you exactly where the risks are.

---

## Managing Identity with Ping Identity MCP

Instead of manually initiating offboarding via multiple dashboards—one for accounts, one for groups, and another for linked applications—you use `delete_user`. This single command handles

This gives you reliable control. Your agent executes complex identity changes that used to require a team of people hours of manual intervention.

the entire process, revoking tokens, nullifying scopes, and purging credentials simultaneously.

---

# Ping Identity: 10 Identity Management Tools

Use these tools to audit user profiles, review application scopes, and execute complex identity lifecycle changes through natural conversation.

#	TOOL	DESCRIPTION
01	<code>delete_user</code>	Permanently deletes a user identity and purges all associated credentials and session tokens from the directory.
02	<code>get_application</code>	Retrieves the security configuration, callback URLs, and grant types for one specific federated application.
03	<code>get_group</code>	Displays all explicit details about a standard Ping Group used to assign permissions.
04	<code>get_user</code>	Retrieves the complete context and metadata for any specified user in the directory.
05	<code>list_applications</code>	Lists all Web, Native, or SPA applications currently federated under your PingOne environment for exposure review.
06	<code>list_groups</code>	Provides a list of identity Groups used to map high-level permissions across SSO channels.
07	<code>list_populations</code>	Lists isolated user populations, such as contractors or B2B clients, which have separate security rules.
08	<code>list_users</code>	Paginates through and lists all internal and external user identities within the standard PingOne Environment.
09	<code>list_risk_policies</code>	Lists active Risk Management rules that trigger real-time multi-factor authentication or block suspicious logins.
10	<code>list_sign_on_policies</code>	Lists logical Sign-on flows and strict authentication conditions, detailing required MFA prompts or complex password structures.

---

## See It in Action

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

**U** Can you list all OIDC applications configured in PingOne and highlight any insecure callbacks?



You currently have 8 configured applications. I noticed that 'Legacy CRM' uses an Implicit Grant and lacks PKCE. Also, 'Staging Dashboard' has a callback URI using a non-secure protocol ( `http://` ). I recommend modifying these components directly in the admin console to maintain Zero-Trust compliance.

**U** Provide the active list of Zero-Trust risk policies governing my environment.



There are 3 active Risk Policies in your designated execution environment.

1. **Impossible Travel** — triggers mandatory MFA prompts upon anomalous geographic location jumps.
2. **Anonymous Networks** — drops authentication directly for identified VPN and TOR entry-points.
3. **Behavior Anomalies** — triggers a step-up challenge utilizing Ping's machine-learning baseline models.

What would you like to drill down into?

**U** Delete the specific suspended user profile assigned to the ID `81773-XYZ-192`.



Execution successful. The PingOne distinct identity assigned to `81773-XYZ-192` has been removed explicitly. Associated explicit MFA linkages, secondary sessions, isolated SCIM references, and assigned app scopes have all been safely purged from the Directory.

---

# Frequently Asked Questions

---

## 01 Can I use Ping Identity MCP to check if an application is secure?

Yes, you can audit the security configuration for any federated app using ``get_application``. This shows whether it uses secure grants or if it has vulnerable callback URIs.

---

## 02 How do I find all external user accounts with Ping Identity MCP?

You use the ``list_populations`` tool. This function lists isolated populations, letting you see groups like 'Contractors' or 'B2B Clients' that are separate from your main employee directory.

---

## 03 What is the best way to manage user roles with Ping Identity MCP?

You should use ``list_groups`` to map out all available identity groups. Then, you can use these group names when checking specific users via ``get_user``.

---

## 04 Is this MCP only for viewing data, or can I delete accounts?

This MCP handles both reading and writing. You can view policies using ``list_risk_policies``, but you also have the power to run the critical ``delete_user`` tool when an employee leaves.

---

## 05 How does Ping Identity MCP handle MFA checks?

You review active policy rules that dictate MFA requirements by calling ``list_risk_policies``. This tells you if a specific login attempt triggers extra authentication steps based on context.







---

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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
 <b>Claude AI</b>	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
 <b>Cursor</b>	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
 <b>VS Code</b>	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add <code>"ping-identity": { "url": "..."</code>
 <b>Windsurf</b>	MCP Settings → <code>mcp_settings.json</code> → Add endpoint URL
 <b>ChatGPT</b>	Settings → Tools & plugins → Add MCP server → Paste endpoint
 <b>Gemini</b>	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

# Ping Identity 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](https://cloud.vinkius.com) →

[vinkius.com](https://vinkius.com) · [support@vinkius.com](mailto: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 Ping Identity. 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	Ping Identity MCP
Server ID	019d75f3-6cfc-7210-8c4a-e062e2800371
Platform	Vinkius Cloud for AI Agents
Endpoint	<a href="https://edge.vinkius.com/{token}/mcp">https://edge.vinkius.com/{token}/mcp</a>

### 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/ping-identity](https://vinkius.com/mcp/ping-identity).