MCP SERVER | NO CODE | CLOUD HOSTED

Pingdom MCP

Monitor Site Uptime and Performance Instantly

Pingdom connects website monitoring and performance data directly into your AI agent. List all uptime checks, track average response times, see historical outages, and manage alerts using natural conversation. Get a complete picture of site health without opening a dashboard.

A+ Quality Score 100/100

uptime-monitoring | performance-tracking | website-reliability | alerts | response-time

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

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.

Pingdom MCP

10 tools available
Cloud-hosted on Vinkius

You can connect Pingdom to any AI client and talk about your infrastructure status like you're talking to a teammate. Instead of jumping between dashboards or running API scripts, you simply ask for the data you need—like listing all active monitoring checks or figuring out why performance dropped yesterday. Your agent handles it immediately. If you want to know what specific probes Pingdom uses worldwide, just ask. It pulls that location list right up. You can even manage maintenance by pausing a check or resuming it with a simple command. All this deep oversight is available through the Vinkius catalog, meaning whatever AI client you prefer, your monitoring data arrives in plain text for instant action.

Core Capabilities

01 — List all site checks

See every current uptime check configured and its real-time status (up, down, or unconfirmed).

02 — Analyze performance history

Get the average response time for any specific check and review detailed outage records.

03 — Audit raw logs

Retrieve raw results or individual check logs to investigate latency spikes or specific errors.

04 — Manage monitoring status

Pause an uptime check for scheduled maintenance, and then resume it when you're ready.

05 — Check global coverage

List all Pingdom probe locations to understand exactly where your site is being monitored globally.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your Pingdom API Token.

02 Connect the MCP to your preferred AI client, like Claude or Cursor.

03 Ask your agent a natural language question about site status, performance, or alerts.

The bottom line is you get real-time website monitoring data and control over checks without touching a web dashboard.

Built For

This MCP serves DevOps Engineers who are tired of logging into multiple dashboards to check site health at 2 AM. It's for Site Reliability Engineers needing instant performance metrics, and System Admins who need a quick way to manage checks during maintenance windows.

DevOps Engineer

Needs to quickly audit recent outages or verify if a service is down after a deployment failure.

Site Reliability Engineer (SRE)

Monitors response time trends and checks global probe locations directly from the IDE without leaving their workflow.

System Administrator

Pauses uptime checks during planned maintenance periods to avoid false alerts, then resumes them when done.

What Changes When You Connect

01 Instantly check site status: Use the `list_uptime_checks` tool to get a complete snapshot of every service's current up/down status without clicking through any dashboard.

02 Analyze performance trends: Get immediate access to average response times using `get_average_response_time`, helping you spot gradual slowdowns before they become critical incidents.

03 Manage maintenance easily: System Admins can run `pause_uptime_check` and then later use `resume_uptime_check` via conversation, eliminating manual state changes.

04 Deep dive into errors: Need to know why a check failed? Use `list_check_results` to pull raw logs and investigate specific latency spikes or error messages instantly.

05 Understand global coverage: Running `list_pingdom_probes` lets you confirm exactly which geographic locations are monitoring your site, ensuring reliable oversight.

Real-World Applications

Checking post-deployment health

A DevOps Engineer just pushed a major update. Instead of opening the Pingdom dashboard and clicking 12 different checks one by one, they ask their agent: 'What's the status of all core services?' The agent uses `list_uptime_checks` and immediately reports which critical APIs are UP or DOWN.

Investigating a slow checkout process

An SRE notices users complain about slowness. They ask their agent for the performance data, triggering `get_average_response_time` on the checkout API check. The agent replies: 'The average time is 800ms; it spiked from normal at 2 PM UTC.' This immediately directs the investigation.

Scheduling planned downtime

A System Administrator needs to update a database and knows the site will be offline for two hours. They use their agent to run `pause_uptime_check` on all non-essential services, preventing false alerts during the maintenance window.

Auditing an intermittent failure

An engineer is debugging a flaky connection issue. Instead of sifting through complex logs, they ask their agent to `list_check_results` for the last 24 hours. The agent pulls raw data showing exactly which time slot experienced high latency.

Patterns to Avoid

Checking status manually

✗ AVOID

Logging into Pingdom and clicking through dozens of checks to build a spreadsheet of current uptime statuses. This takes 20 minutes, and the data is already stale by the time you finish.

✓ INSTEAD

Just ask your agent: 'List all active uptime checks.' The agent uses `list_uptime_checks` and gives you the real-time status list in seconds.

Confusing logs for performance

✗ AVOID

Seeing a log error message (e.g., HTTP 503) but not knowing if it represents a permanent outage or just a temporary blip.

✓ INSTEAD

Don't rely on raw data alone. Ask the agent to `get_check_outages` for that check ID; this provides historical context and confirms if the issue was an actual service failure.

Ignoring global coverage

✗ AVOID

Assuming your site is monitored equally well from all countries, only to find out a key region isn't covered by any Pingdom probe.

✓ INSTEAD

Always run `list_pingdom_probes` first. This confirms the exact geographical spread of monitoring and helps you plan coverage gaps.

The Right Fit

Use this MCP if your primary need is to turn complex, multi-tab web dashboards into simple conversational queries. If you are an SRE or DevOps team that needs immediate visibility into uptime status, average response times, or outage history, this is for you. However, don't use it if you only need basic ping checks; while Pingdom handles those, this MCP excels at combining multiple functions—like listing probes *and* checking performance metrics in one chat session. If your goal is pure code generation or complex database manipulation unrelated to site health, then a dedicated data connector will be better.

The Manual Pain of Site Monitoring

Right now, checking your website's health means jumping through hoops. You open the Pingdom dashboard, click on 'Checks,' and then you have to manually scroll or filter for every service you care about. If a site is slow, you might need to jump to another tab just to find the average response time. Then, if there was an outage days ago, you're dealing with a whole different section that requires specific IDs and clicks. It’s tedious copy-pasting and dashboard hopping.

With this MCP integrated into your agent, all that manual clicking vanishes. You simply ask: 'What are the performance metrics for our core services?' The agent pulls together status, average response times, and even historical outage data—all in one conversation thread. It's instant, structured intelligence.

Get Site Health Status with Pingdom

You don’t have to remember which tool name controls what. Instead of hunting through documentation for the right API call, you just tell your agent: 'Give me the status.' It handles fetching the list of all active checks using `list_uptime_checks` and gives you a clear answer.

The result is simple: your AI client processes complex monitoring data and hands you actionable text. You stop managing dashboards and start making decisions.

Pingdom with 10 Tools

These tools let you query every aspect of your website's monitoring setup—from listing all active checks to getting raw historical error logs.

#	TOOL	DESCRIPTION
01	`get_check_details`	Retrieves detailed information for a specific website check.
02	`get_check_outages`	Lists historical outages that occurred for a given monitoring check.
03	`get_average_response_time`	Calculates and retrieves the average response time for any specific check.
04	`list_uptime_checks`	Provides a list of all active uptime monitoring checks configured in Pingdom.
05	`list_alert_contacts`	Shows the contacts who receive alerts when a site goes down.
06	`list_maintenance_windows`	Lists all scheduled maintenance windows currently defined in Pingdom.
07	`list_pingdom_probes`	Provides a list of every monitoring location (probe) Pingdom uses globally.
08	`list_check_results`	Retrieves raw, individual results or logs for specific check runs.
09	`pause_uptime_check`	Temporarily stops an uptime check from running until manually reactivated.
10	`resume_uptime_check`	Restarts a previously paused uptime check, getting it back into the monitoring cycle.

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 current uptime checks and their status.

I've retrieved 10 checks. 'Main Site' is UP, 'Checkout API' is UP, but 'Legacy Support Portal' is currently DOWN. Would you like to see the error log for the down service?

U What was the average response time for the 'Main Site' check (ID: 12345) today?

The average response time for 'Main Site' today is 342ms. Performance has been stable with a slight spike to 580ms around 10:00 AM UTC.

U Pause the uptime check for ID 98765 for our scheduled maintenance.

Check 98765 has been successfully paused. Remember to ask me to resume it once the maintenance is finished!

Frequently Asked Questions

01 **How do I check my uptime status using the Pingdom MCP?**

You list all active checks by asking your agent to use `list_uptime_checks`. This immediately tells you if critical services are up, down, or unconfirmed.

02 **Can I find out why my site was slow last week with Pingdom MCP?**

Yes. You ask the agent to run `get_check_outages` for a specific service ID. This tool provides a clear record of past failures and when they occurred.

03 **What if I need to pause monitoring during maintenance? How does Pingdom MCP help?**

You instruct the agent to run `pause_uptime_check` on the relevant service ID. This prevents false alerts and keeps your system running smoothly until you tell it to resume.

04 **Does Pingdom MCP show me where my site is monitored?**

Absolutely. Run `list_pingdom_probes` to get a comprehensive list of every physical location worldwide that monitors your site, confirming global coverage.

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 `"pingdom": { "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

Pingdom 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 Pingdom. 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	Pingdom MCP
Server ID	`019d75f3-4b24-73ec-82fc-38de7d3bc67b`
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/pingdom.