MCP SERVER | NO CODE | CLOUD HOSTED

Kisi MCP

Manage locks, users, and physical access control instantly.

Kisi MCP connects your AI agent directly to cloud-based access control systems, allowing you to manage locks, users, and physical security settings using natural conversation. You can instantly check door statuses across multiple locations, trigger remote unlocks for emergencies, and audit who has permission to enter specific areas.

A+ Quality Score 100/100

physical-access-control | smart-locks | cloud-security | user-management | remote-access | facility-management

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

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.

Kisi MCP

9 tools available
Cloud-hosted on Vinkius

Managing building access used to mean logging into a dashboard, clicking through menus, and running reports—a huge time sink when you're dealing with an emergency or a simple status check. Now, your agent handles that overhead. You just tell your AI client what needs doing: 'Unlock the main entrance for the delivery crew.' The MCP executes it instantly. It gives your agent the power to handle everything from listing all doors and checking if they are online to querying complex user roles across entire organizations. All this capability is available through Vinkius, making connection simple. You can delegate tasks like reviewing who belongs to which access group or getting detailed information on a specific door without touching a web interface.

Core Capabilities

01 — Remote Door Control

You send an unlock command and the assigned door opens, regardless of your physical location.

02 — Device Status Check

The agent lists all managed locks (doors) and reports their real-time status—online/offline or locked/unlocked.

03 — User Profile Retrieval

You ask for a person's details, and the MCP pulls their full profile information from Kisi.

04 — Permission Audit

The system checks complex access rules by listing roles assigned to specific groups or places.

05 — Location Mapping

You list and inspect every physical location (place) configured in your security environment.

One Click on Vinkius — From Prompt to Execution

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

01 First, subscribe to this MCP on Vinkius and input your Kisi API Key (Personal Access Token).

02 Next, connect your preferred AI client to the catalog. Your agent now sees all available security tools.

03 Finally, you give a simple text prompt—like 'List all locks that are currently offline'—and the MCP executes the command.

The bottom line is that you talk to your AI client, and it talks directly to your building's physical lock system.

Built For

Facility Managers who spend too much time manually checking door status across multiple buildings. Security Technicians who need immediate remote access or user role audits during an incident. Operations Engineers managing complex, distributed properties.

Facilities Manager

You use the MCP to list physical places and check which doors are reporting issues across your portfolio without opening a single dashboard.

Security Coordinator

You quickly perform remote unlocks or audit user roles to verify credentials when an incident happens, bypassing slow manual processes.

DevOps Engineer

You integrate door status monitoring and access control logic directly into custom scripts, relying on the MCP for reliable data streams.

What Changes When You Connect

01 You don't need to switch between tabs. Instead of manually checking the status of every door in a building via a web dashboard, you simply ask your agent to list all locks, getting real-time data on online/offline status.

02 Emergency response time shrinks dramatically. Instead of calling maintenance and waiting for them to physically open a door, you tell your AI client to unlock_door immediately using the specific lock ID.

03 User onboarding is faster. You can ask the agent to list users and then review their full profiles via get_my_profile without having to navigate through complex user directories.

04 Auditing becomes trivial. To check if a team has too many permissions, you just query groups and places using list_access_groups and list_role_assignments instead of running multi-page reports.

05 Location oversight is simplified. You can use list_places to get an inventory of all sites across your portfolio at a glance, making facility audits much faster.

Real-World Applications

Emergency Access Required

A security guard needs immediate access to the server room and knows the door ID. Instead of calling maintenance or waiting for keycard validation, they prompt their agent: 'Unlock the Server Room door (ID: 9876).' The MCP executes unlock_door instantly.

Weekly Compliance Audit

A facilities manager needs to know which employees are currently active and what access groups they belong to. They instruct their agent to list users, followed by calling list_role_assignments to verify compliance status across the company.

System Health Check

An operations engineer suspects several doors aren't communicating. They prompt for 'list all locks', immediately seeing which devices are reporting as offline, allowing them to dispatch repair crews instantly.

Site Expansion Planning

A team needs to map out a new wing. They use list_places to see the current site structure and get details for a specific location using get_place_details before planning where new access points are needed.

Patterns to Avoid

Checking status manually

✗ AVOID

Logging into the Kisi dashboard, selecting 'Locks,' then clicking through dozens of door entries to see if they're all online and locked.

✓ INSTEAD

Just ask your agent to list locks. This runs a single command that checks every device's real-time status for you.

Finding user details

✗ AVOID

Using the search bar on the Kisi website, entering an employee name, and then having to click through several permission tabs to see their role.

✓ INSTEAD

Ask your agent to list users first. Then you can retrieve full profile information using get_my_profile for targeted data.

Managing permissions

✗ AVOID

Trying to figure out if a group has access because the documentation is too complex, requiring manual cross-referencing of groups and places.

✓ INSTEAD

Query role assignments using list_role_assignments. This tool consolidates all permission logic into one actionable report.

The Right Fit

Use this MCP if your primary bottleneck involves real-time physical access management—if checking door status, managing users, or physically opening a lock is a recurring task that requires jumping between different dashboards. It's essential for operations teams who need immediate, conversational control over their facility infrastructure. Don't use it if you are only performing deep data analysis (like long-term audit log review). For pure historical record keeping, a dedicated logging tool will be better; this MCP focuses on live status and actionable commands like unlock_door or list_locks. If your need is simply to view the static details of one component, use get_lock_details. But if you need an overview, stick with listing tools.

The Chore of Physical Access Management

Right now, checking your building's security means a painful click-fest. You have to log into the access control portal, navigate to 'Devices,' then manually check dozens of locks one by one. If you need to audit who can enter an area, you run reports on users, then cross-reference groups and places, copy-pasting data between tabs until your eyes glaze over.

With this MCP, the entire process shifts from clicking to talking. You simply ask your agent for a status report or to check permissions. The AI client handles all the dashboard navigation and data collation behind the scenes. You get an immediate, clean answer—no more endless reports.

Kisi: Immediate Access Control Management

Tasks like checking every device's status or running a user list used to require multiple logins and segmented views. You had to use different sections of the dashboard just to get an inventory, then another section to see role assignments.

Now, you delegate that complexity. You ask your agent to consolidate everything—from listing all locks to checking which users are assigned roles—and it delivers one integrated answer. The physical security infrastructure finally talks directly to your conversational workflow.

Kisi: 9 Access Control Tools

These tools let your AI client interact directly with your building's security system, allowing you to check lock statuses, manage user profiles, and audit access groups.

#	TOOL	DESCRIPTION
01	`get_lock_details`	Retrieves specific operational information about a single door lock.
02	`get_place_details`	Fetches detailed configuration data for a designated physical location or site.
03	`list_access_groups`	Generates a comprehensive list of all predefined access groups used in your system.
04	`list_locks`	Lists every managed lock (door) in the facility, providing an overview of all devices.
05	`get_my_profile`	Retrieves the profile details for the user currently authenticated with Kisi.
06	`list_places`	Lists every physical place or location configured within your security management system.
07	`list_role_assignments`	Shows all active role assignments, allowing you to monitor who has what level of permission.
08	`unlock_door`	Sends an immediate command to remotely unlock a specific door lock.
09	`list_users`	Provides a complete list of all users registered in the Kisi organization for management purposes.

See It in Action

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

U Unlock the 'Main Entrance' door (ID: '12345') in Kisi.

I've sent the unlock command to 'Main Entrance'. The door should now be unlocked.

U List all locks that are currently offline.

I've checked your devices. Currently, 'Side Exit (ID: 6789)' and 'Storage Room (ID: 1011)' are reporting as offline.

U Show me the details for the place 'Headquarters'.

I've retrieved the details for Headquarters (ID: 9876). It currently has 12 active locks and 5 access groups associated with it.

Frequently Asked Questions

01 **How do I remotely unlock a door using the Kisi MCP?**

You use the unlock_door tool by providing the specific lock ID. Your agent sends the command to the lock, and it should open immediately.

02 **Does Kisi MCP let me see all users in one place?**

Yes, you can list all registered users using list_users. This provides a full directory of accounts in your organization.

03 **Can I check if a specific location is set up correctly?**

You use get_place_details to retrieve the configuration details for any given physical place ID, ensuring it meets compliance standards.

04 **What if I need to know which groups exist in my system?**

Call list_access_groups. This tool gives you a complete inventory of all access groups defined within your Kisi environment.

05 **How do I find out what a user's current roles are?**

You query the system using list_role_assignments. This shows every role assigned to every group, helping you audit permissions quickly.

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 `"kisi": { "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

Kisi 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 Kisi. 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	Kisi MCP
Server ID	`019d75c1-06af-7095-9944-829ea2ebdeba`
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/kisi.