MCP SERVER | NO CODE | CLOUD HOSTED

Autho MCP for AI Agents

Manage User Identities and Client Access Programmatically

Auth0 allows your AI agent to take full control of complex identity infrastructure. You can manage user accounts, audit security logs, review client applications, and configure roles—all through natural conversation with any MCP-compatible client.

A+ Quality Score 98.33/100

authentication | authorization | sso | user-management | mfa | identity-provider

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/auth0-alternative

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.

Autho MCP

13 tools available
Cloud-hosted on Vinkius

Managing an enterprise's user identities shouldn't require opening a dozen browser tabs just to check one setting. With this MCP, your AI agent acts like a dedicated identity operations engineer. You connect your Auth0 tenant once and gain programmatic control over every aspect of your user base.

Instead of clicking through dashboards, you tell your agent what you need. Want to find all users who signed up using GitHub? Done. Need to check if the 'Finance' client application is still active? Your agent handles it. It lets you list and modify roles, audit connection settings (like Google or SAML), or review security logs for failed logins. When your internal tools are spread across different platforms, Vinkius makes sure your AI can access everything in one place. You manage user lifecycles, application configurations, and identity connections—all without leaving the chat window.

Core Capabilities

01 — Manage User Accounts

Create new users, retrieve full profiles by email or ID, update metadata, or permanently delete accounts.

02 — Audit Security Activity

Review global tenant logs and specific user activity logs to track login attempts, password changes, and API operations.

03 — Monitor Applications and Clients

List all registered client applications (web apps, mobile apps) and view the connections used for authentication (Google, GitHub, SAML).

04 — Control Roles and Permissions

View defined roles and their associated permission sets to audit or confirm Role-Based Access Control (RBAC) configurations.

05 — Examine Organizational Structure

List all multi-tenant organizations configured within your Auth0 tenant for B2B visibility.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/auth0-alternative** — connect your AI agent in three steps.

01 Subscribe to this MCP and provide your Auth0 domain and Management API Token.

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

03 Instruct your agent: 'Find all users who failed to log in last week,' or 'List all connected identity providers.' The agent executes the command directly.

The bottom line is, you stop navigating dashboards and start having conversations with your security infrastructure.

Built For

This MCP targets technical roles that spend too much time context-switching between security consoles. Security teams need to audit quickly; DevOps needs programmatic control over users and applications. If you regularly manage who can log in and what permissions they have, this is for you.

Security Engineer

Audits failed login attempts or reviews all identity connections to ensure no unauthorized sign-in methods are active.

DevOps Engineer

Manages users programmatically, checks client applications for outdated credentials, and lists roles to enforce least privilege access.

Product Manager

Monitors user growth by listing all registered users or reviews organization membership status before a major feature launch.

What Changes When You Connect

01 Audit failed logins instantly. Use `list_logs` to see specific event types, like distinguishing between a wrong password failure ('f') versus an invalid email address ('fu').

02 Control user lifecycles entirely via your agent. You can use `create_user` and `update_user` without ever logging into the Auth0 console.

03 Audit application permissions efficiently. By running `list_clients`, you get a full inventory of every web app or SPA that authenticates users, helping to spot forgotten integrations.

04 Simplify user lookups. Instead of guessing IDs, use `get_user_by_email` to find a user profile regardless of which connection (Google, database) they signed up with.

05 Maintain compliance easily. You can run `list_roles` and review the entire RBAC structure, ensuring that roles are correctly defined and assigned.

Real-World Applications

Investigating a Security Breach

A security team notices suspicious activity. They ask their agent to run `list_logs` for the last 24 hours, filtering by IP address and event type. The agent returns all API operations that match the criteria, identifying the source of the breach instantly.

Onboarding a New Product Line

A DevOps engineer needs to onboard a new single-page application (SPA). They use `get_client` after generating a temporary client ID, confirming that the configuration allows for necessary callbacks before deploying code.

Updating User Contact Info at Scale

Product management needs to change the default theme metadata for 50 users. They use `list_users` with a search query and then instruct their agent to run `update_user` on all results, setting the new JSON object data.

Auditing Identity Providers

A compliance officer asks for a list of all authentication methods. The agent uses `list_connections`, providing an immediate overview of every integrated provider like Google or GitHub, along with their configuration status.

Patterns to Avoid

Using the GUI for bulk changes

✗ AVOID

A user tries to manually update 20 users' roles and metadata by clicking through the UI one by one, which is slow and prone to error.

✓ INSTEAD

Tell your agent to execute `list_users` first, then pass the resulting list of user IDs back into a single `update_user` command. This handles bulk changes programmatically.

Confusing client types

✗ AVOID

A developer gets confused about whether they need to check the general application settings or if the specific API key is already listed in the connection audit.

✓ INSTEAD

First, run `list_connections` to see all integrated providers. Then use `get_client` with the client ID to confirm that the correct credentials are associated.

The Right Fit

Use this MCP if your primary pain point is coordinating identity changes across multiple security consoles and dashboards. You need programmatic access to user data, audit logs (`list_logs`), and application configurations (`list_clients`). Don't use it just because you want a better user interface; the value is in the *action*. If all you do is read information without taking an action (like creating or deleting), you might be fine with other documentation tools. But if you need to execute commands—like `create_user` or `update_user` —this MCP handles that complexity for your agent.

Autho MCP for AI Agents: Solving Complex User Identity Management

Today, managing user accounts in an enterprise setup is a nightmare of tabs. You have to jump between the 'Users' dashboard, the 'Audit Logs,' and the 'Client Applications' section just to figure

With this MCP, you don't navigate; you talk. Your agent handles the context switching for you. You simply ask your AI client to 'Show me all users whose accounts were created in the last month

out why a login failed or who changed a permission set last week. Copy-pasting IDs and manually reviewing dozens of log entries takes hours.

who failed a login attempt.' The result is immediate, structured data that tells you exactly what happened and why.

Autho MCP for AI Agents: Auditing Authentication Connections

Manually auditing identity connections means checking if every required provider—be it Okta, Google, or a custom database—is configured correctly and hasn't been orphaned. You have to cross-reference multiple settings pages just to confirm the source of truth.

Now, your agent runs `list_connections`. It immediately provides a definitive list showing every connected identity provider and its current status. That single output gives you full visibility across your entire authentication stack.

Manage 13 Tools for User Identity and Access Auditing

Use these tools to perform specific actions on users, clients, roles, and security logs within your Auth0 tenant.

#	TOOL	DESCRIPTION
01	create_user	Creates a brand new account for a user using the required connection type and email address.
02	delete_user	Permanently deletes an Auth0 user, removing all associated data and logs. This action cannot be undone.
03	get_client	Retrieves specific details about a registered client application using its unique ID.
04	get_user	Fetches the profile and metadata for any specific user, whether they used Google or their personal credentials to sign in.
05	get_user_by_email	Finds all users associated with a single email address, which is helpful if multiple connection types are involved.
06	list_clients	Lists every client application registered in your tenant, showing their type and allowed callbacks for auditing purposes.
07	list_connections	Shows all identity providers (like Okta or GitHub) that users can sign into, useful for security audits of login methods.
08	list_logs	Generates a comprehensive list of security and activity logs across the entire tenant, allowing filtering by event type or date range.
09	list_organizations	Lists all multi-tenant organizations you have set up for B2B models, including their branding details.
10	list_roles	Retrieves a list of defined permission sets (RBAC roles), which is key for auditing access controls.
11	list_user_logs	Focuses on activity logs specific to one user, helping troubleshoot account-level issues or review behavior over time.
12	list_users	Searches and lists all users in your tenant using advanced filtering by name or email address.
13	update_user	Modifies specific fields of an existing user, such as updating their email or changing their blocked status.

See It in Action

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

U Show me all users who failed to log in today, specifically listing the IP address for each failure.

Security Audit Report: Failed Logins (Today)

- **Count:** 17 failed logins.
- **Failure Breakdown:**

* Invalid Email ('fu'): 5 attempts. Top emails: `user@corp.com` (3), `test@corp.com` (2).
* Wrong Password ('f'): 12 attempts. Most common IP range: 203.0.113.x.

Key Insight: The majority of failures originated from the same internal subnet, suggesting a potential brute-force attempt on accounts with known usernames.

U Find all client applications that are no longer in use or have outdated credentials.

Client Application Review (Need Action)

The following clients may need review:

CLIENT ID	NAME	TYPE	LAST USED DATE	STATUS
`abc-123`	Legacy Widget App	SPA	9 months ago	Active (Review Required)
`xyz-789`	Internal Dashboard	Web	Never	Dormant/Unused

Please check if the 'Legacy Widget App' needs to be disabled.

U Create a new user account for john.doe@company.com and give them admin access.

Success: User `john.doe` (ID: auth0|abc123xyz) has been created.

Details:

- Email Verified: Yes
- Status: Active
- Roles Assigned: ['admin', 'editor']
- Notes: The new user is fully configured and ready to use the platform.

Frequently Asked Questions

01 **How can I audit my user base when I don't know their User IDs using Auth0 MCP for AI Agents?**

You don't need the ID. You can use your agent to search and list all users by providing just an email address or a partial name. The system returns the full profile details, letting you see everything about that account.

02 **Can I programmatically manage user roles with Auth0 MCP for AI Agents?**

Yes. You can list all existing permission sets using the role tools and then update a specific user's profile to assign or remove permissions without clicking anything.

03 **What if I need to check if an application is still connected to Google OAuth?**

You first run a list of connections to see all identity providers. Then, you can use the tools to get details for specific client applications, confirming which credentials are linked.

04 **Is Auth0 MCP for AI Agents useful for checking user activity and logins?**

Absolutely. You can generate detailed security reports by listing logs. This lets you filter massive amounts of data down to just the failed login attempts or API calls that matter most.

05 **Can I delete a user account completely using this MCP?**

Yes, but be careful. You can use the dedicated tool to permanently delete an account. Remember, this is irreversible, so always double-check who you're targeting before confirming.

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 `"auth0-alternative": { "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

Autho 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 Auth0. 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	Auth0 MCP
Server ID	`019d8419-6f37-73bb-8b90-47fc2b640407`
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/auth0-alternative.