MCP SERVER | NO CODE | CLOUD HOSTED

ApplicantStack MCP for AI Agents

Manage your entire recruiting and job pipeline from one place

ApplicantStack brings your entire hiring lifecycle into your AI agent. You track job openings, manage candidate profiles, and move people through custom workflow stages—all without leaving your chat interface. It handles everything from initial job listing retrieval to updating a candidate's status from 'Interviewing' to 'Hired.'

A+ Quality Score 100/100

applicant-tracking | hiring-workflow | onboarding | job-postings | candidate-management | recruitment-automation

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

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.

ApplicantStack MCP

7 tools available
Cloud-hosted on Vinkius

This MCP connects your recruiting data directly into your AI workspace. Instead of jumping between ApplicantStack and multiple spreadsheets, you tell your agent what you need—like, 'Show me all candidates who interviewed last week but haven't been updated.' The agent handles the complexity; it pulls candidate records, checks job metadata, and even moves people to new workflow stages when they pass review.

It lets you manage everything from job listings to onboarding data. You can list every open role or drill down into a specific person’s profile for an instant summary. If your team uses other AI clients, you'll find this MCP listed in the Vinkius catalog, giving you one place to connect all your enterprise tools.

Core Capabilities

01 — Review Candidate Profiles

Retrieve full details for any candidate using their ID or name.

02 — Track Open Jobs

List all your current and closed job postings, or pull deep metadata on a specific role.

03 — Audit Applicant Database

See an aggregate list of all candidates in the system, allowing filtering by workflow stage or score.

04 — Advance Candidate Status

Instantly change a candidate's position within your defined hiring workflow stages.

05 — Generate Onboarding Lists

Access data for new hires, ensuring you have all the necessary information to start their employee journey.

One Click on Vinkius — From Prompt to Execution

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

01 You prompt your AI agent with a natural language request (e.g., 'List all candidates currently in the Interview stage').

02 The MCP processes the request, calls the necessary internal tools to query ApplicantStack's data.

03 Your agent receives structured results—whether it's a list of open jobs or a candidate's updated profile—and presents it back to you in plain text.

The bottom line is that your AI client acts as the middleman, translating natural conversation into specific data actions within ApplicantStack.

Built For

This MCP is critical for recruiters and HR teams drowning in systems. If you spend time copying candidate names from one sheet to another, or constantly switching between job board views and internal CRMs, this tool saves hours of clicking.

Recruiter

Quickly checking the status of candidates across multiple roles without opening a dozen tabs.

Hiring Manager

Getting AI-assisted summaries on candidate profiles and performance scores before an interview loop.

HR Administrator

Tracking hiring trends across the entire company or ensuring all onboarding tasks are initiated for new hires.

What Changes When You Connect

01 Instead of manual cross-referencing, you can ask the agent to pull data on a candidate's progress across multiple jobs using 'list_candidates' and 'get_job'.

02 When a candidate moves stages (e.g., Interview to Offer), calling 'update_candidate' automatically adjusts their profile status in your system.

03 HR teams can instantly review onboarding requirements by running 'list_hires', guaranteeing no new employee falls through the cracks.

04 Hiring managers get immediate context on open roles by using 'list_jobs' and quickly checking job metadata without logging into a separate portal.

05 You avoid the data silos problem. All candidate details, from initial application to final hire status, are accessible via one command.

Real-World Applications

A Candidate Needs Status Check

A recruiter needs to know if a specific applicant is still in the running. They ask their agent for the candidate's details using 'get_candidate'. The agent returns an up-to-date summary, confirming they are currently in the 'Final Review' stage.

Listing All Open Roles

A hiring manager needs to plan Q3 staffing. They prompt their agent to list all available jobs using 'list_jobs'. The agent delivers a clean, sortable spreadsheet of open roles and their required metadata.

Automating Stage Progression

The interview panel finishes reviewing Jane Doe's profile. Instead of logging into ApplicantStack, the manager asks the agent to update her status using 'update_candidate', moving her directly from 'Interview' to 'Offer Pending'.

Reviewing Recent Hires

The HR team needs a list of everyone who started last month. They use 'list_hires', getting an immediate manifest that includes required onboarding documentation links for every new employee.

Patterns to Avoid

Manually updating candidate stages

✗ AVOID

The recruiter logs into the ApplicantStack portal, finds John Doe's profile, clicks the dropdown menu to change his stage from 'Interview' to 'Hired', and then manually updates his email in a separate CRM.

✓ INSTEAD

Simply use your agent. Tell it: 'Move candidate John Doe to the Hired stage.' The MCP handles the stage update via 'update_candidate' and can be prompted to push that data across connected systems.

Forgetting job metadata

✗ AVOID

The hiring manager sees a listing for 'Marketing Director,' but has no idea who owns it or what salary range was approved. They have to email HR just to ask.

✓ INSTEAD

Use the agent's ability to get full job details by calling 'get_job'. You instantly pull metadata like ownership, department code, and budget constraints right in your chat.

Missing new hire data

✗ AVOID

The HR admin runs a report on hires but misses the required tax form submission for one person because the system interface is confusing.

✓ INSTEAD

Run 'list_hires' through the MCP. The agent summarizes all necessary onboarding tasks for every new employee, ensuring nothing is skipped.

The Right Fit

Use this MCP if your team needs to centralize candidate data and workflow actions within a single conversational interface. It’s perfect for recruiters who need to list jobs, check specific profiles, or automate status changes (like moving someone from 'Interview' to 'Offer').

Don't use it if you just need simple document storage—you'll need a dedicated file management tool instead. Also, this MCP focuses on the *state* of the candidate record; if you need complex analytics (e.g., predicting turnover rates), you’ll need to connect it to a separate BI or data warehousing service.

This is for managing the process flow itself. You're asking 'What stage are they at?' or 'Show me all open roles,' not 'Analyze this dataset to predict X.'

ApplicantStack MCP: Solving Candidate Tracking Pain Points in Recruiting

Today, managing a single candidate is a nightmare. You start on one system for the job description, switch tabs to another platform for their resume, and then jump into a third CRM just to see their interview notes. Every time you have to copy an ID or check a status across systems, you lose minutes of productive time.

With this MCP, those manual hops disappear. You simply ask your agent to pull up the full profile using 'get_candidate'. The system gathers all that metadata—job details, workflow stage, and hiring history—and presents it in one clean summary. It's instant context.

ApplicantStack MCP: Automating Hiring Workflow from Job Listings

Before this tool, launching a new role meant manually updating the job board, notifying internal

Now, you tell your agent to list all jobs using 'list_jobs'. You get not only the open position but

stakeholders, and creating tickets in three different project management tools. The process was slow, error-prone, and required coordination across multiple departments.

also the associated metadata—like ownership or budget code—right then. It makes launching roles fast and verifiable.

7 Tools in the ApplicantStack MCP for Candidate Management

These tools let your agent access every part of your hiring data, from listing all jobs to updating a single candidate's status.

#	TOOL	DESCRIPTION
01	`get_account_check`	Verifies that your connection to ApplicantStack is active and ready to use.
02	`get_candidate`	Retrieves detailed information for a single candidate record.
03	`get_job`	Pulls specific data points about one job listing.
04	`list_candidates`	Generates an overview list of all applicants in the system.
05	`list_hires`	Provides a manifest of all records flagged as new hires, including onboarding data.
06	`list_jobs`	Fetches a list containing every active and closed job listing in your account.
07	`update_candidate`	Updates candidate information, most commonly used to change their stage in the hiring workflow.

See It in Action

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

U List all active job openings for the Engineering department.

Active Job Listings: Engineering

- **Software Engineer II:** Open since 2024-10-01. Location: Remote. Owner: Jane Doe. Status: Active.
- **DevOps Specialist:** Open since 2025-01-15. Location: NYC. Owner: John Smith. Status: Needs Review.
- **Lead Architect:** Open since 2024-08-20. Location: Remote. Owner: Jane Doe. Status: Active.

U What's the status and details for candidate ID C567?

Candidate Profile: Alex Johnson (C567)

- **Current Stage:** Offer Pending
- **Last Activity:** 2026-06-10 (Status updated by Agent)
- **Source Job:** Software Engineer II
- **Summary:** Highly skilled in Python and cloud infrastructure. Passed all technical rounds with high marks.

U Move candidate 'C998' to the Hired stage.

✅ Success. Candidate C998 has been successfully moved from 'Interview' to **Hired**. The system automatically logged this change and flagged the record for HR onboarding review.

Frequently Asked Questions

01 **How do I find my ApplicantStack API Token?**

Log in to ApplicantStack, go to **Settings**, then **Edit Settings**. Your API token will be listed under the API section.

02 **What is the subdomain?**

The subdomain is the first part of your ApplicantStack URL (e.g., if your URL is `mycompany.applicantstack.com`, your subdomain is `mycompany`).

03 **Can I move a candidate to a new stage?**

Yes, use the `update_candidate` tool and provide the new stage name in the `stage` field to advance them in your workflow.

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 `"applicantstack": { "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

ApplicantStack 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 ApplicantStack. 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	ApplicantStack MCP
Server ID	`019d7550-7e6b-7040-9bd2-8557ab15fa4a`
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/applicantstack.