MCP SERVER | NO CODE | CLOUD HOSTED

# Assembly MCP for AI Agents

## Managing professional service client and company records

The Assembly MCP connects your AI agents directly to your professional service records. You can retrieve full details on clients, companies, users, workspaces, and historical notes all through natural conversation. Need to know a client's history or pull current company data? Your AI agent handles the lookup so you don't waste time clicking through dashboards.

A+ Quality Score 100/100

client-management | workspace-management | crm | professional-services | data-retrieval

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

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.

# Assembly MCP

10 tools available
Cloud-hosted on Vinkius

This MCP lets your AI agents access every piece of structured data inside your Assembly platform. Instead of logging into the app and manually navigating between tabs to find a client's record, their associated companies, or past notes, you simply ask your agent what you need. It pulls together everything: from listing all active clients to getting deep details on a specific workspace setup. You can also look up individual users or check historical documents with `get_note`. If you're building an automated workflow that relies on accurate, current client information, this MCP is the connective tissue. Vinkius hosts this connector so your AI agent—whether it's Claude, Cursor, or another compatible client—can access all these records from one place and use them to answer complex business questions.

## Core Capabilities

**01 — List all client records.**

Provides a list of every active client in your platform.

**02 — Get specific client details.**

Retrieves the full record for one identified client.

**03 — List all company records.**

Provides a list of every associated company across your clients.

**04 — Get specific company details.**

Fetches the full record for one identified company.

## One Click on Vinkius — From Prompt to Execution

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

01 Connect your Assembly account to Vinkius using an API key. This gives your AI agent permission to read your professional service data.

02 Your AI client sends a natural language request, like 'What was the last note for Acme Corp?'

03 The MCP translates that request into specific tool calls (e.g., `get_note`) and returns the structured record directly to your agent.

The bottom line is you get Assembly's complex data exposed as simple, actionable function calls your AI agent can run.

## Built For

This MCP is essential for consultants and account managers whose job requires pulling together a client's full service history across multiple internal systems. If you spend time copying data from one tab to another, this tool saves hours.

**Account Manager**

Pulls up all related records—clients, companies, and notes—to give a client an accurate status update during a review call.

**Sales Operations Specialist**

Builds automated reports that pull together data from `list_clients` and associated company details to flag accounts needing immediate attention.

## What Changes When You Connect

01 Instantly retrieve a client's full history. Instead of jumping between tabs, you can ask the agent to pull details using `get_client` or `list_clients`, giving you one consolidated view.

02 Avoid manual data entry for reporting. By listing all companies with `list_companies`, your AI agent pulls structured lists that feed directly into reports and summaries.

03 Stay organized on team knowledge. You can search historical context by calling `get_note` or using `list_notes`, ensuring no vital project detail is lost in email chains.

04 Consolidate project scope details. Use `list_workspaces` to get an overview of all active projects and immediately dive into specific records with `get_workspace` when needed.

05 Get a complete picture of the user base. You can list all users ( `list_users` ) or check individual permissions using `get_user`, which is key for compliance reviews.

# Real-World Applications

**A client asks about their account status.**

Instead of digging through emails, the agent can quickly run `list_clients` to confirm the account exists, then use `get_company` and `get_workspace` to list all associated services and active projects in one response.

**Onboarding a new team member.**

A manager needs to understand who has access to what. The agent uses `list_users` for an overview, then runs `get_user` on specific roles to verify permissions and details.

**Preparing for a quarterly business review (QBR).**

The agent pulls together key data points by running `list_clients`, then calling `list_notes` and selecting the three most recent notes to summarize historical interactions before the meeting.

# Patterns to Avoid

### Treating all data as a single source.

✗ AVOID

Asking the agent a vague question like 'Tell me about this client' which forces it to guess which company or workspace you mean, resulting in incomplete answers.

✓ INSTEAD

Be specific. Always ask for the full scope: 'List all clients and retrieve the most recent note for Acme Corp.' This directs the agent to use `list_clients` and then `get_note` correctly.

### Ignoring associated records.

✗ AVOID

Only asking for client details using `get_client`, but failing to mention that the client has multiple companies or workspaces, leaving out critical context.

✓ INSTEAD

Always prompt for related data. Ask: 'What are all the companies and active workspaces linked to Client X?' This ensures both `list_companies` and `list_workspaces` run.

### Assuming a universal view of notes.

✗ AVOID

Asking, 'Give me notes for this quarter,' without specifying which workspace or client the note belongs to. The agent might return irrelevant data.

✓ INSTEAD

Specify context every time. Use: 'List notes only within the Consulting Ops workspace.' This uses `list_notes` while scoping it down using the workspace ID.

## The Right Fit

Use this MCP if your job involves cross-referencing multiple, distinct records—for example, matching a client's details to their associated company and checking the latest note on that specific project. You need comprehensive data retrieval across the board.

Don't use it if you just need a simple list of names; for that, most general directory tools work fine. However, if your process requires pulling complex relationships—like knowing which user is tied to which workspace and what company they are associated with—then this MCP is necessary because it handles the deep data connections inherent in professional services.

---

# Assembly MCP for AI Agents: Managing Professional Service Client Records

Today, when you need to build a status report or give a client an update, you spend time clicking through tabs. You check the main client file, then switch over to the 'Associated Companies' tab. Next, you jump into the 'Workspaces' section to see active projects. Then, you have to manually find and copy key details from notes that were created weeks ago.

With this MCP, those steps disappear. Your AI agent executes a single command: it gathers all related data—the client, their companies, their workspaces, and the specific notes—and presents it in one coherent output. You get immediate, comprehensive context without leaving your chat window.

---

# Assembly MCP for AI Agents: Tracking Company and User Data

Manually tracking user access or company affiliations is a nightmare. When you need to know which users have permissions in a specific workspace, or what companies are linked to a client that isn't the primary one, you usually run

This MCP centralizes those lookups. You can instantly list all users and then use `get_company` alongside the user data to verify affiliations. It makes auditing roles and corporate structures fast and verifiable.

separate manual checks across different sections of the platform.

# Assembly 10 Tools for Client & Company Data Retrieval

Use these tools to list every record type—clients, companies, users, workspaces, notes—and retrieve specific details on demand.

| # | TOOL | DESCRIPTION |
|---|---|---|
| 01 | `get_client` | Retrieves the full record for a single, specified client. |
| 02 | `get_company` | Fetches all details related to one specific company entity. |
| 03 | `get_note` | Looks up the content and metadata for a single, identified note. |
| 04 | `get_user` | Gets detailed information about an individual user account. |
| 05 | `get_workspace` | Retrieves all details connected to a specified workspace. |
| 06 | `list_clients` | Generates a list of every client record available in Assembly. |
| 07 | `list_companies` | Lists all associated company records across the platform. |
| 08 | `list_notes` | Provides an index or list of all existing notes. |
| 09 | `list_users` | Generates a comprehensive list of all user accounts. |
| 10 | `list_workspaces` | Lists every available workspace within Assembly. |

# See It in Action

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

**U** I need a summary of Acme Corp's status, including their main company details, active workspaces, and any notes from last month.

**Acme Corporation Status Report**

- **Company:** Global Tech Solutions (Active since 2018)
- **Workspaces:** 3 Active Projects:
* Product Dev: Members include J. Smith, M. Jones.
* Consulting Ops: High activity, last updated yesterday.
* Finance: Currently reviewing Q3 budget.
- **Recent Notes:**
* 'Q2 Revenue Forecast': Discussed revenue goals (Updated 2024-06-10).
* 'Client Onboarding Checklist': Confirmed next steps with the client.

**U** List all clients and tell me which ones have more than two associated companies.

I ran a full scan of your client roster. Here are the findings:

- **Vinkius:** Has 4 associated companies (Primary, Beta, Dev Ops, Legal).
- **Google:** Has 2 associated companies.
- **BetaStream:** Has 3 associated companies (Core, Mobile, Cloud).

These records give you a clear view of your client structure at a glance.

U Show me the details for user Sarah Jenkins and all workspaces she is involved in.

**User Profile: Sarah Jenkins**

- **Role:** Senior Consultant (ID: 456)
- **Last Login:** Today, 10:30 AM.
- **Workspaces Involved:**

1. Product Dev (Lead Contributor)
2. Finance (Reviewer)
3. Client Onboarding (Admin)

# Frequently Asked Questions

01 **How does the Assembly MCP help me pull a client's complete service history?**

It aggregates all related records—clients, companies, workspaces, and notes—into one place. Instead of clicking through multiple tabs, your agent pulls everything together so you get immediate context on their full relationship with you.

02 **Do I need to worry about missing data when listing clients using the Assembly MCP?**

No. The tool uses the API key connection to read all structured data available in your platform, ensuring that lists like `list_clients` and `list_companies` are comprehensive.

03 **Can I use this for internal team organization too?**

Yes. You can list users and workspaces to get an overview of who is working on what, helping you quickly audit roles and project scope without manually checking user permissions.

04 **What kind of notes or documents can I retrieve with the Assembly MCP?**

You can pull historical context. The agent allows you to list all notes across the account and get specific notes by retrieving them, which is key for summarizing past conversations during a call.

05 **Is this better than just using native dashboard filtering?**

Yes. Dashboards show what's visible on screen; this MCP lets your agent execute complex queries across siloed data points (like matching a client to a note and a company) that might not be displayed together in one view.

06 **If I change my API key, will the Assembly MCP still work?**

Yes, as long as your AI client has valid access credentials linked through Vinkius, it can maintain connection to the underlying data structure. You just need to update the keys in the system.

# 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 `"assembly": { "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

# Assembly 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 Assembly. 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 | Assembly MCP |
| Server ID | `019dd0bd-10b0-73c0-b9cc-06c94ef08a83` |
| 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/assembly.