

MCP SERVER

NO CODE

CLOUD HOSTED

# Zoho Desk MCP

Manage support tickets and contacts naturally.

Zoho Desk MCP is an AI-powered connection that gives your agents full control over support operations. Your agent can manage customer tickets, update contacts, and track complex issues using natural conversation. It lets you list open tickets across departments, assign priorities, add internal notes for team context, and reply directly to the customer through one interface.

**A+** Quality Score 100/100

ticket-management

support-automation

helpdesk

customer-service

issue-tracking

internal-notes



# 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.

### 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.

### 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.

### 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**

Encryption at rest

**Ed25519**

PKI vault signatures

**24h TTL**

Ephemeral session keys

**V8 Isolate**

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.

### 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.

### 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.

### 06 — Honeypot Trap System

Phantom credentials are injected into isolated environments. If a honeypot 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.

**FinOps**

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

**Firewall & DLP**

PII redaction activity, sensitive data protection counters, and security event timeline.

**Agent Activity**

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

**Tool Health**

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

**Incident Log**

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

Get started at [cloud.vinkius.com](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Zoho Desk MCP

13 tools available

Cloud-hosted on Vinkius

This MCP turns Zoho Desk into a smart, AI-ready support center. Instead of manually navigating dashboards, your agent handles the whole workflow using plain language commands. You can ask it to find all high-priority tickets related to billing or search for a specific product issue across departments. If you need to talk to a customer before starting work, your agent first pulls up their details and account history. Need to update the ticket? It handles changing status, assigning new owners, and documenting internal notes so the whole team stays in sync. When you connect this MCP through Vinkius, your AI client gains access to every support function—from creating a brand-new contact record to closing out an old one. This means your agents don't just read tickets; they manage the entire customer journey autonomously.

---

## Core Capabilities

### 01 — Search and list all open tickets

Find specific support issues, or view a complete overview of every ticket across all departments.

### 02 — Manage contacts and accounts

Look up customer details and create new records for people who haven't interacted with the system before.

### 03 — Update ticket status and ownership

Change a ticket from open to resolved, or reassign it to another specific agent within the department.

### 04 — Communicate publicly or privately

Send a reply visible to the customer through email, or add confidential notes only your team can see.

### 05 — Create new tickets quickly

Generate an entirely new support ticket with proper subject matter, priority level, and department assignment.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/zoho-desk](https://vinkius.com/mcp/zoho-desk) — connect your AI agent in three steps.

- 01 Connect your Zoho Desk account using the OAuth token from the Zoho API Console.
- 02 Provide the necessary Organization ID found in your Zoho Desk setup screens.
- 03 Tell your AI agent what you need—for example, 'Find all open billing tickets for John Doe and add an internal note that we are escalating this issue'.

The bottom line is: your AI client accesses the full suite of support tools without you having to write a single API call.

---

## Built For

This MCP is essential for customer success managers, IT helpdesk admins, and support team leads. If you spend half your day copying IDs between tabs or manually updating ticket statuses, you need this. It lets your agents handle the full lifecycle of a customer issue so you can focus on solving complex problems.

### Support Manager

Uses it to list departments and search tickets quickly to see if resources are correctly allocated or if specific areas need more staff.

### IT Helpdesk Administrator

Manages the entire ticket lifecycle, using this MCP to create new tickets, assign them priorities, and track their progress from open to closed.

### Customer Success Manager

Looks up customer details using `get_contact` before replying, ensuring they have all necessary account information right away.

---

## What Changes When You Connect

- 01 Stop searching for IDs. Use `list_departments` first, then use that ID to create a ticket or assign an agent, keeping your workflow linear.

- 
- 02 You can review full history before responding. Calling `get_ticket` gives your agent all the context needed so they don't have to ask the customer basic questions.

---

  - 03 Never lose internal context again. The `add_note` tool lets you document private thoughts and escalations that stay off the customer's radar.

---

  - 04 Handle complex relationships easily. `List accounts` helps you understand which company a contact belongs to, giving your support reps crucial B2B context instantly.

---

  - 05 Speed up triage. Instead of reading everything, use `search_tickets` to find exactly what you need by keyword across all open issues.
- 

---

## Real-World Applications

### **A customer calls with a complex issue and needs immediate assignment.**

The agent asks the system to 'Find the right department for this billing payment failure.' The MCP uses `list_departments`, then `create_ticket`, ensuring the ticket gets assigned immediately to Finance rather than general support.

### **A new client signs up and needs a support ticket opened.**

The agent first uses `create_contact` with the new user's details. Then, they immediately use `create_ticket` for them, setting the priority to high since it's a brand-new, critical account.

### **A follow-up is needed on an old, complex bug report.**

The agent needs context. They use `get_ticket` for the relevant ID, review all past replies and notes, and then uses `add_note` to remind their manager about the next steps before replying publicly using `add_reply`.

### **A team needs to know who is available right now.**

The support manager uses `list_agents` combined with `list_departments` to see which agents are currently free and best suited for handling the sudden influx of tickets.

---

# Patterns to Avoid

---

## Only using search tools

### ✗ AVOID

Running `search_tickets` repeatedly just to gather basic info like a contact's name or department ID.

### ✓ INSTEAD

First, run `list_contacts` or `get_contact` to verify the user details. Then use this validated information when you create a ticket or assign an agent.

---

## Mixing public and private updates

### ✗ AVOID

Adding critical internal notes about payment issues into the visible reply field for the customer.

### ✓ INSTEAD

Always use `add_note` for team-only thoughts. Use `add_reply` only when you are ready to send a message that must be seen by the client.

---

## Ignoring organizational structure

### ✗ AVOID

Trying to create tickets or assign agents without knowing which departments exist.

### ✓ INSTEAD

Start with `list_departments`. This gives your agent all the necessary IDs so you can properly assign the ticket and department.

---

## The Right Fit

Use this MCP if your support process involves multiple steps: looking up customer info, creating records (contacts/tickets), assigning ownership, documenting internal notes, and then communicating a final reply. It is perfect for teams that need an AI agent to act as a full-time Tier 1 support employee.

Don't use this if your only goal is simple data retrieval; tools like `list_tickets` or `search_tickets` are enough. But if you need the agent to *do* something with that data—like creating a ticket, updating its status, or adding notes for later review—this MCP is what you need.

---

---

## The headache of switching between support tabs and spreadsheets.

Right now, dealing with an incoming customer request means clicking through five different tabs: one for the contact's profile, another to see their past tickets, a third to check who is available in that department, and finally, jumping back to the ticket to manually type status updates or copy-paste notes for your team. It's slow, it requires constant context switching, and you lose time just finding the right button.

With this MCP, the process changes completely. You tell your agent: 'Check John Doe's account, find his open billing tickets, reassign them to Finance, add a note saying I spoke to him today, and then reply to him.' The AI handles all those clicks in one conversation.

---

## Zoho Desk MCP gives your team full control over the support lifecycle.

Tasks like creating a ticket are complex. You can't just write 'make a ticket.' You have to know the correct department ID, check if the contact exists, and decide on priority before you use `create_ticket`. This friction slows down everything.

Now, your AI agent takes those hidden steps away. It handles all that logic in the background, letting you focus entirely on the customer conversation.

---

# Zoho Desk MCP with 13 Tools

These tools allow your AI client to perform specific actions within Zoho Desk, letting you manage everything from listing departments to adding private notes.

#	TOOL	DESCRIPTION
01	add_note	Adds a private note visible only to agents for internal team context or escalation details.
02	add_reply	Sends a formal reply that the customer can see via email and in their portal.
03	create_contact	Registers a completely new person into Zoho Desk, capturing basic details like name, email, and phone number.
04	create_ticket	Generates a brand-new support ticket, setting its subject, contact, department, and required priority.
05	get_contact	Retrieves all known details about a specific customer or contact record.
06	get_ticket	Fetches the full history and current status of any existing support ticket for review.
07	list_accounts	Lists all associated organizational accounts, useful for understanding which company a customer belongs to.
08	list_agents	Checks the availability and details of agents within a specific department.
09	list_contacts	Retrieves a list of existing customer contacts, including their account association and phone number.
10	list_departments	Lists all departments used for categorizing tickets and assigning staff, providing necessary IDs for other tools.
11	list_tickets	Returns a list of support ticket summaries across the entire organization by status or priority.
12	search_tickets	Searches ticket subjects and descriptions using keywords to find related issues quickly.
13	update_ticket	Modifies an existing ticket, changing its status or reassigning it without altering the whole history.

---

## See It in Action

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

**U** List all open high-priority tickets from the Support department



I'll search for high-priority open tickets in your Support department right away.

**U** Create a new ticket for john@example.com about billing issue with subject 'Payment Failed'



I'll create the billing ticket and assign it to the appropriate department.

**U** Add a public reply to ticket #12345 saying 'We are investigating your issue and will update you shortly'



I'll post the reply to the ticket right now.

---

## Frequently Asked Questions

**01** How do I check if a department exists before assigning a ticket with Zoho Desk MCP?

First, use `list_departments` to pull all available departments. This gives you the required unique IDs necessary for creating or updating tickets.

**02** Can Zoho Desk MCP only handle open tickets?

No. You can use `get_ticket` to review full ticket history, even if it was previously closed, giving you complete context before responding.

---

**03 Does creating a contact with Zoho Desk MCP require all fields?**

No. While some details are mandatory, the tool allows setting optional data like phone numbers or department associations when you run `create_contact`.

---

**04 What is the difference between `add_note` and `add_reply` using Zoho Desk MCP?**

Add an internal note for comments visible only to your team. Use `add_reply` specifically when you need to send a message that will appear in the customer's view.

---

**05 Does Zoho Desk MCP help me find which agents are available?**

Yes, `list_agents` lets you check agent names and their current availability status within a specific department, helping with proper assignment.







---

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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
 <b>Claude AI</b>	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
 <b>Cursor</b>	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
 <b>VS Code</b>	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add <code>"zoho-desk": { "url": "..." }</code>
 <b>Windsurf</b>	MCP Settings → <code>mcp_settings.json</code> → Add endpoint URL
 <b>ChatGPT</b>	Settings → Tools & plugins → Add MCP server → Paste endpoint
 <b>Gemini</b>	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

# Zoho Desk 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](https://cloud.vinkius.com) →

[vinkius.com](https://vinkius.com) · [support@vinkius.com](mailto: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 Zoho Desk. 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	Zoho Desk MCP
Server ID	019d762a-1853-73ff-9f44-55ef06b8f37a
Platform	Vinkius Cloud for AI Agents
Endpoint	<a href="https://edge.vinkius.com/{token}/mcp">https://edge.vinkius.com/{token}/mcp</a>

### 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/zoho-desk](https://vinkius.com/mcp/zoho-desk).