MCP SERVER | NO CODE | CLOUD HOSTED

LibreTranslate API MCP

Automate Language Detection and Translation Audits

LibreTranslate API allows your agent to handle complex localization and language auditing tasks using free, open-source data. You can ask your AI client to translate text into dozens of languages, detect the primary language of any string, or audit a content catalog for missing translations—all through natural conversation. It gives you instant linguistic insight without ever opening a dedicated translation website.

A+ Quality Score 100/100

machine-translation | language-detection | open-source | api-integration | text-processing

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

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.

LibreTranslate API MCP

4 tools available
Cloud-hosted on Vinkius

When working with multilingual content, checking consistency and making sure the right words land in the right cultures is usually a headache. This MCP lets your agent manage that entire process. You don't need to jump through separate portals or maintain dozens of API keys for different services. Instead, you talk to your AI client, and it handles the complexity. It can instantly translate text between multiple languages, check which language a piece of content is written in, and even list every language supported by the service. This means whether you're researching regional market copy or managing complex documentation, your agent acts like an immediate linguistic consultant. By connecting this MCP through Vinkius, you get reliable access to all these translation capabilities, keeping your workflow focused on writing, not API calls.

Core Capabilities

01 — Detect text language

The tool figures out the primary language of any given piece of text.

02 — Translate content instantly

It translates text from a source language into a specified target language.

03 — List supported languages

You can get a complete catalog of all languages the service supports.

04 — Check API operational status

The tool confirms if the translation service is currently working correctly.

One Click on Vinkius — From Prompt to Execution

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

01 First, connect your preferred AI client to this MCP and optionally enter your specific LibreTranslate API Key or custom instance URL.

02 Next, you instruct your agent using natural language commands, asking it to perform a translation, check the language of a document, or list supported codes.

03 Your agent executes the request through the necessary tool calls and returns the translated text, the detected language code, or the complete metadata.

The bottom line is that your AI client handles all the technical setup; you just tell it what translation work needs doing.

Built For

Content teams and developers who are constantly dealing with international content, localization QA, or market expansion. This MCP helps prevent costly mistakes caused by inconsistent language usage or poor detection.

Technical Writer

Uses the agent to verify that all localized strings in a document match required regional styles and languages before publication.

Localization Manager

Automates language audits across entire content libraries, confirming which markets are covered and what translation data is available.

Developer / Engineer

Integrates reliable, open-source translations into Python scripts or development pipelines without writing complex API wrapper code.

What Changes When You Connect

01 You get instant language detection. Instead of guessing or manually checking a document's source language, the agent uses `detect_language` to confirm it immediately.

02 Stop worrying about API maintenance. You connect once via Vinkius and gain access to translation capabilities without needing to manage multiple vendor credentials.

03 Run bulk content audits easily. By calling `translate_text`, you can quickly check semantic distribution across dozens of target languages for consistency checks.

04 Gain full linguistic visibility. The `list_supported_languages` tool gives you a complete catalog, helping you plan which markets to localize next.

05 Reliable operational status is guaranteed. Using `check_api_status` ensures your agent never fails mid-workflow because the translation service went down.

Real-World Applications

Auditing a multilingual website's content

A content manager needs to verify if 50 pages have been properly translated into German. They ask their agent to run the `detect_language` tool on all page bodies, then use `translate_text` repeatedly with the target language set to German, immediately flagging any missing or incorrectly coded articles.

Preparing for international product launch

A technical writer is drafting documentation and needs to confirm if their platform supports Japanese. They use the agent to call `list_supported_languages`, confirming support, and then test a sample string using `translate_text` to ensure proper cultural phrasing.

Analyzing research data from diverse sources

A researcher collects articles in French, Mandarin, and Portuguese. Instead of manually translating everything, they ask the agent to detect the language for each source using `detect_language`, then translate all non-English snippets into one common format.

Building a multi-market data pipeline

A developer needs to write a script that processes text from three regions. They use the agent to call `translate_text` for each region and then pass all resulting translations into a single structured output, confirming consistency.

Patterns to Avoid

Using generic translation widgets

✗ AVOID

Copying text from a website to Google Translate, getting the result, and then manually re-pasting it into documentation. This is slow, error-prone, and requires constant context switching.

✓ INSTEAD

Use this MCP to let your agent handle the process end-to-end. Your agent calls `translate_text` directly, integrating the translation result immediately into your workflow without manual copy/paste.

Relying on vague language tools

✗ AVOID

Using a simple 'auto-detect' tool that only gives a general category (e.g., 'Romance Language') instead of the specific ISO code needed for coding or documentation.

✓ INSTEAD

The `detect_language` tool provides precise identification and confidence scores, giving you the exact language code you need to move forward with development.

Forgetting about service outages

✗ AVOID

Building a critical daily workflow that breaks entirely when an external translation API goes offline for maintenance or rate limits.

✓ INSTEAD

Always start your automated process by calling `check_api_status` to confirm the connection is live. This prevents workflow failures and saves you time.

The Right Fit

Use this MCP if your primary pain point involves managing, auditing, or translating content across multiple human languages, especially in technical documentation or large-scale data sets. If you need to confirm the source language of a string, use `detect_language`. If you need to convert text into a target market's specific dialect, use `translate_text`. Don't use this if you just need simple word definitions; for that, a basic dictionary tool is better. You should also consider using `list_supported_languages` early in your project planning phase to ensure the service can handle all necessary language pairs.

Dealing with global content means constant context switching and manual checks.

Right now, if you have a piece of documentation that needs to go out globally, your process looks like this: You copy the English source text into one tool; you paste it into a second service for translation into Spanish; then you check the resulting Spanish against the original requirements in a third tab. It's slow, and every time you manually paste or switch tabs, there’s a chance of error.

With this MCP, your agent eliminates those steps entirely. You ask your AI client to translate content from English to Japanese, for example. The request is processed by the agent using `translate_text`, and it delivers the final, translated text directly back to you in one conversation thread. It’s immediate and reliable.

You get instant linguistic audit reports with LibreTranslate API MCP.

Manually verifying language consistency across a large content library is impossible. You have to check the source text, then verify the translation for every single entry, and then confirm the target market's dialect—it’s a nightmare of tabs and spreadsheets.

Now, you let your agent run `detect_language` against your entire corpus. It reports back instantly, showing exactly which language each piece of content is in. You get reliable data that lets you focus on refining the message, not fixing the metadata.

LibreTranslate API: 4 Tools for Translation and Detection

These tools allow you to programmatically check service status, detect the source language of text, list all supported languages, or translate content between any two specified languages.

#	TOOL	DESCRIPTION
01	`check_api_status`	Checks if the LibreTranslate service is currently running and available for use.
02	`detect_language`	Identifies the language of a specific block of text you provide.
03	`list_supported_languages`	Retrieves a comprehensive list of all languages supported by this instance.
04	`translate_text`	Translates text from one language into another using the open-source translation service.

See It in Action

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

U Translate 'Hello world' from English to Portuguese using LibreTranslate.

I've translated the text for you! 'Hello world' in Portuguese is identified as 'Olá mundo'. Would you like the full JSON response or a different target language audit?

U Detect the language of the text 'C'est la vie'.

I've analyzed the text! It is identified as French (fr) with a confidence score of 99%. I can provide the full metadata list of supported languages if you'd like to confirm.

U List all languages supported by LibreTranslate.

I've retrieved the language catalog! There are dozens of languages supported, including English, Spanish, French, and Chinese. I can help you translate text between any of these thematic markers.

Frequently Asked Questions

01 **How does LibreTranslate API MCP handle different dialects?**

The service supports translation between dozens of languages and helps audit semantic distribution. You can use `translate_text` to test how a phrase translates into specific regional variations.

02 **What is the difference between detecting language and listing supported languages with LibreTranslate API MCP?**

Detecting language (`detect_language`) analyzes *a piece of text* you provide. Listing supported languages tells you *what languages* the entire system can handle.

03 **Do I need to worry about paying fees when using LibreTranslate API MCP?**

This tool uses open-source data, which means it is designed for reliable, cost-effective usage in professional workflows. Check the specific instance URL provided by your team.

04 **Can I use LibreTranslate API MCP to translate code snippets?**

The service is built for natural language text and documentation; while you can submit code comments, it's optimized for linguistic content, not functional programming syntax.

05 **Is the connection stable? How do I check LibreTranslate API MCP status?**

You should always run `check_api_status` at the start of your session. This verifies that the service is operational before you submit any valuable text for translation.

06 **Is an API Key required for LibreTranslate API?**

No. LibreTranslate is open-source and many public instances allow free requests. However, some instances or the official site may require an API Key for higher limits.

07 **Can I use a custom self-hosted instance?**

Yes. Provide the `LIBRETRANSLATE_URL` in the server configuration to point your agent to your own private instance.

08 **Does it support language detection?**

Yes. Use the `detect_language` tool to identify the primary language of any text string along with a confidence metadata score.

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 `"libretranslate-api": { "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

LibreTranslate API 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 LibreTranslate API. 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	LibreTranslate API MCP
Server ID	`019d8453-4e2e-7381-87e7-b5426a6988d1`
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/libretranslate-api.