

MCP SERVER

NO CODE

CLOUD HOSTED

# WhatsApp Message Sender MCP

## Send Alerts and Confirmations Directly to WhatsApp

The WhatsApp Message Sender MCP lets your AI agents send simple text messages directly to any WhatsApp number using Meta's official Cloud API. It's a secure, one-way communication bridge for sending automated alerts, confirmations, or status updates without needing complex chatbots or full inbox access.

**A+** Quality Score 100/100

messaging

notifications

api-integration

customer-engagement

alerts

webhook



# 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.

# WhatsApp Message Sender MCP

1 tools available

Cloud-hosted on Vinkius

You don't need a complicated chatbot system just to tell a customer their order shipped. This MCP gives your agent something much simpler: the ability to send a pure text message via WhatsApp. Your AI client can drop payment confirmations, emergency alerts, or simple status updates straight into a user's chat, bridging a gap between your internal systems and one of the world's biggest messaging platforms.

Because this tool is strictly for sending messages using Meta's official API endpoint, it operates like a secure megaphone. Your agent can fire off an alert without ever having to read or reply to anything in the user's inbox. You connect this MCP through Vinkius and your AI client gets immediate access to direct customer communication. This makes it perfect for operational tasks—when an agent processes an order, it doesn't just log it; it immediately pings the user on WhatsApp with the outcome.

---

## Core Capabilities

### 01 — Send Status Alerts

Your AI client can send simple text messages to customers when a specific process finishes, like an order being shipped or an account being updated.

### 02 — Confirm Payments and Orders

The agent sends automated alerts confirming successful payments or providing details about completed transactions directly through WhatsApp.

### 03 — Broadcast Emergency Notifications

You can use your AI client to dispatch immediate, time-sensitive messages to a group of users for critical updates or warnings.

# One Click on Vinkius — From Prompt to Execution

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

- 01** First, you connect your preferred AI client to the WhatsApp Message Sender MCP on Vinkius.
- 02** Next, instruct your agent what needs to be sent (the message body) and who it goes to (the destination phone number).
- 03** The agent executes the necessary tool call, sending the text through the official Meta API instantly.

The bottom line is that you tell your AI client what to send and where, and it handles the secure delivery using WhatsApp's established communication channel.

---

## Built For

Anyone whose job involves giving real-time updates or confirmations to a customer base. Think operations managers who get frustrated having to manually copy confirmation numbers into emails, or support agents who need instant proof that an alert was sent.

### Operations Manager

Uses the tool to ensure customers receive immediate alerts—like 'Your package is out for delivery'—the moment a physical process changes.

### Customer Success Specialist

Sends automated follow-up messages confirming that documentation has been received or services have been activated, closing the loop with the client.

### Support Engineer

Dispatches critical service outage notifications to affected users immediately after an incident is resolved, bypassing email delays.

---

## What Changes When You Connect

- 01** Immediate Customer Reach: When an agent finishes processing a request, it sends the outcome directly on WhatsApp. No more waiting for emails to be read.

- 
- 02** Pure Containment: This MCP only allows sending messages; your AI client cannot read replies or mess with business settings. It's secure and strictly one-way.

---

  - 03** Zero Integration Bloat: You skip massive SDKs. The tool uses the direct REST API endpoint, keeping your setup lean and focused purely on delivery.

---

  - 04** Operational Clarity: Instead of relying on ticket status changes, you send a concrete alert that confirms action taken—like an order processing completion.

---

  - 05** Reliable Notifications: It handles critical alerts like payment confirmations or service downtime notices through the most widely used messaging app available.

---
- 

---

## Real-World Applications

### Order Status Update

A customer asks their agent, 'Where's my stuff?' The agent doesn't have to open a tracking portal; it calls the tool and sends WhatsApp message saying that the package is out for delivery.

### System Failure Alert

A monitoring tool detects a critical service outage. The operations team feeds that data into their workflow, and the agent sends out an emergency WhatsApp message to all affected users.

### Payment Confirmation

An accounting system completes a transaction. Instead of waiting for manual email confirmation, the agent uses this MCP to send an immediate 'Your payment went through' text alert.

---

# Patterns to Avoid

---

## Assuming Two-Way Chat

### X AVOID

The user thinks they can use this MCP to create a conversational chatbot that reads replies and maintains context.

### ✓ INSTEAD

This MCP is a megaphone, not a mailbox. It only sends messages. If you need the agent to read incoming replies or manage conversation history, look for an inbox-reading category tool instead.

---

## Ignoring Service Windows

### X AVOID

The agent tries to send a promotional message outside of WhatsApp's 24-hour user window.

### ✓ INSTEAD

Remember the Meta API rules. For messages sent outside standard service hours, you must use pre-approved templates; otherwise, the tool call will fail.

---

## Overcomplicating the Goal

### X AVOID

Trying to send a complex document or image attachment via a simple text message API.

### ✓ INSTEAD

This MCP is restricted to pure text messages. If you need rich media, check out other communication APIs that support file uploads; this tool only handles plain body text.

---

## The Right Fit

Use this MCP if your primary goal is reliable, one-way notification and alerting—you just need to tell someone something simple, like 'Your ticket was resolved' or 'The report is ready.' Don't use it if you expect the agent to engage in conversation, answer follow-up questions, or read replies. For two-way communication, search for a dedicated messaging platform MCP that allows inbox access. If your goal is just data transfer and logging, an internal database connector will work better.

---

---

## The Manual Chore of Customer Alerts

Today, when a key operational task finishes—like processing a complex payment or updating inventory—someone has to manually write the update. They copy the confirmation number from one dashboard, paste it into an email template, and then hit send. This process is slow, error-prone, and creates a frustrating delay between action and customer awareness.

With this MCP, the system handles the final mile. The agent executes the task and immediately sends the final alert via WhatsApp. You get instant confirmation of the outcome, eliminating manual copy-pasting and making your operational feedback loop immediate.

---

## WhatsApp Message Sender: Direct Delivery

The biggest pain point that vanishes is the need to manage multiple notification channels. You don't have to worry about whether the customer checked their email or if they opened a text message; you just send it through WhatsApp.

Your AI client now has a direct, single-purpose communication line to customers, ensuring critical status updates arrive instantly and reliably.

---

# WhatsApp Message Sender: 1 Tool

These tools let you programmatically send plain text notifications and alerts to any WhatsApp number via a secure, one-way connection.

#	TOOL	DESCRIPTION
01	<code>send_whatapp_message</code>	Sends a direct text message to a specified WhatsApp number using the Meta Cloud API. Note that sending outside of the 24-hour service window requires pre-approved templates.

---

## See It in Action

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

**U** Send a WhatsApp message to 5511999999999 saying that their order is out for delivery.



I've successfully dispatched the WhatsApp message.

---

## Frequently Asked Questions

### 01 Can the WhatsApp Message Sender MCP read my replies?

No. This MCP is strictly a sending tool. Your AI client can only dispatch messages; it cannot access your inbox, view conversation history, or reply to incoming messages.

### 02 What format must the destination number be for `send_whatapp_message`?

The phone number must be provided in E.164 format and crucially, without any plus signs (e.g., 5511999999999). Always verify the recipient's number before sending.

### 03 Is this MCP suitable for marketing campaigns?

Because it's a pure sender, it can be used for targeted alerts. However, remember that Meta requires pre-approved templates for bulk messaging outside of the 24-hour window.

### 04 How is this MCP different from using an email API?

Unlike email APIs, which require recipients to check their inbox, this sends a notification directly into WhatsApp. It's better for time-sensitive alerts where immediate visibility matters most.

### 05 Can the agent read incoming WhatsApp replies with this?

No. This MCP utilizes the REST API strictly for creating new messages. It does not configure or expose incoming webhooks, meaning it acts strictly as a one-way notification megaphone. It cannot see your customer's replies.

## 06 Why are some messages failing to send?

The Meta WhatsApp Cloud API has a strict 24-hour service window. If you are sending a free-form text message to a user who hasn't messaged you in the last 24 hours, the API will block it unless you use a pre-approved Message Template. Ensure you are within the 24-hour window for free-form replies.







---

# 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>"whatsapp-message-sender": { "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

# WhatsApp Message Sender 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 WhatsApp Message Sender. 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	WhatsApp Message Sender MCP
Server ID	019e390a-4bbf-716f-b7b6-a2ca8a0cf971
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/whatsapp-message-sender](https://vinkius.com/mcp/whatsapp-message-sender).