

M C P S E R V E R

N O C O D E

C L O U D H O S T E D

WhatsApp Chat Export Parser MCP for AI Agents

Analyzing message history from WhatsApp chats for key details

WhatsApp Chat Export Parser takes messy, exported WhatsApp chat files and converts them instantly into structured JSON data. It lets your AI search years of conversations, automatically finding lost addresses, summarizing long threads, or calculating participant message statistics—all while keeping the parsing entirely local for privacy.

A+ Quality Score 100/100

chat-export

regex-parsing

data-extraction

message-history

search-optimization

text-analysis



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 — connect your AI agent in under 60 seconds.

WhatsApp Chat Export Parser MCP

1 tools available

Cloud-hosted on Vinkius

Finding a single address from a conversation that happened months ago feels impossible when you're scrolling through thousands of messages. This MCP solves that problem completely. When you export any chat via WhatsApp's built-in feature, you get a dense `.txt` file—a mess to read. Our system parses this whole file locally using advanced regex, pulling out every message with its exact timestamp, sender, and content into clean JSON. Your AI client can then search, filter, or summarize your entire chat history instantly. You don't have to manually sift through years of texts anymore; you just ask the question, and it finds the answer. Because Vinkius hosts this MCP in their catalog, connecting it is simple—you connect once from any compatible agent, and suddenly you can treat all your private conversations like searchable databases.

Core Capabilities

01 — Extracting full message records

The parser reads the chat file to pull out every individual message, noting the sender, content, and precise timestamp.

03 — Analyzing participant statistics

The tool calculates who sent the most messages within a group chat and provides those specific counts.

05 — Processing large chats safely

It manages huge files by sending only the first 100 and last 50 messages plus participant stats, preventing context overflow errors.

02 — Handling dual-locale dates

It automatically identifies and parses date formats, whether they use US style (1/15/23) or European format (15/01/2023).

04 — Summarizing long conversations

Your agent can read massive exports and condense months of dialogue into concise summaries, identifying key topics discussed.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/mcp/whatsapp-chat-export-parser — connect your AI agent in three steps.

- 01 Export a conversation from WhatsApp using its built-in 'Export Chat' feature to get a raw `.txt` file.
- 02 Upload that exported chat file into your AI client connected via this MCP.
- 03 The system runs the data through `parse_whatsapp_chat`, which converts the messy text into structured, searchable JSON.

The bottom line is, you go from a scrolling wall of text to an instantly queryable database record with three simple steps.

Built For

Anyone who uses WhatsApp for communication—whether it's coordinating massive projects or managing family logistics—needs this. If your professional life depends on data hidden inside group chats, you need to connect this MCP.

Project Manager

They use it to analyze large group chat exports and quickly summarize decisions made across multiple threads or identify who was responsible for specific action items.

Real Estate Agent

They parse chats with clients to find lost details, like the exact address of a viewing or an agreement on a showing time that was mentioned weeks ago.

Team Lead / Coordinator

They analyze group chat message history to generate participant statistics, determining which team member contributed the most dialogue or attention in a given timeframe.

What Changes When You Connect

- 01 Stop manually scrolling through thousands of messages. `parse_whatsapp_chat` converts entire chat histories into structured JSON, making everything searchable instantly.

-
- 02 Never miss a detail again. The parser handles both US and EU date formats automatically, ensuring every timestamp is accurate for your AI search.
 - 03 Protect your privacy. All parsing happens locally on your end; nothing leaves your machine and gets uploaded to the cloud.
 - 04 Manage massive exports without crashing your agent context. The system intelligently processes huge chats by focusing on key segments and participant data.
 - 05 Pinpoint specific information—like an agreement or a phone number—regardless of when it was sent, using the structured output from `parse_whatsapp_chat`.
-

Real-World Applications

Finding lost addresses in old chats

An agent parses a chat export and locates a specific address mentioned three months ago. The user asks, 'Where did Maria say the new office was?' and gets an exact quote with the date and sender.

Summarizing project decisions from group chats

A team lead uploads a massive chat export and prompts their agent to summarize all agreed-upon action items for Q3. The agent returns a bulleted list of commitments, saving hours of reading.

Determining message contribution in groups

The user wants to know who dominated the conversation flow. They run participant statistics and get a clear count showing which team member sent the most messages over a two-week period.

Patterns to Avoid

Copying and pasting snippets

X AVOID

Trying to manually search for an address or date by copy/pasting small chunks of text into your agent, which fails if the context is too large.

✓ INSTEAD

Use ``parse_whatsapp_chat`` to process the entire `.txt`` file. Once it's structured JSON, you can ask your agent complex questions across the whole dataset in one go.

Using generic text analyzers

X AVOID

Relying on basic document parsers that struggle with unique conversational formatting or mixed date formats (US vs. EU).

✓ INSTEAD

This MCP uses dual-locale regex and is designed specifically to handle the unpredictable, messy structure of WhatsApp exports.

Overloading agent context memory

X AVOID

Trying to feed an entire 10GB chat export into your AI client at once, causing context crashes or timeouts.

✓ INSTEAD

The system handles massive chats by intelligently sending only the first 100 and last 50 messages plus participant stats, keeping the process stable.

The Right Fit

Use this MCP if your primary goal is to search for specific data points—like names, dates, or addresses—that are buried deep inside a large volume of messy chat logs. If you need to analyze conversation flow (e.g., who talked the most), use it. Don't use this if you only need to check a single message; just copy and paste that one line instead. Also, don't rely on general document parsers for complex data extraction; they can't handle the specific formatting of WhatsApp chats. This MCP is built exclusively around the unique structure of the exported `.txt` file.

WhatsApp Chat Export Parser: Solving Messy Messaging Data Extraction

Right now, finding a single piece of information—like that agreement on rent or that recommendation for a restaurant—means endless scrolling through thousands of lines of text. You copy the file and paste it into your agent, but you can't ask precise questions because the data isn't structured; it's just one big, unstructured wall of text.

With this MCP, you upload the raw chat export, and the system does the heavy lifting by running dual-locale regex. Instead of a block of unreadable text, your agent gets clean JSON records. You can then ask complex questions like, 'What did we decide about the payment date?' and get an instant, sourced answer.

WhatsApp Chat Export Parser: Analyzing Communication Flow Statistics

Before this tool, figuring out who contributed most to a project chat was guesswork. You might manually count messages or rely on imprecise summaries that missed the nuances of group discussion.

Now you can run participant statistics directly through `parse_whatsapp_chat`. It gives you hard numbers showing exactly who sent how many messages and what percentage of the overall conversation flow they represented.



WhatsApp Chat Export Parser: 1 Messaging History Parsing Tool for Data Extraction

Upload chats and use this tool to extract structured records, timestamps, sender identities, and participant statistics from WhatsApp text files.

#	TOOL	DESCRIPTION
01	parse_whatsapp_chat	Uploads an exported WhatsApp .txt file and extracts all messages, senders, timestamps, and participant statistics into structured JSON data.

See It in Action

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

U Parse my WhatsApp chat with Maria and find the message where she sent me an address.



Found it! On 03/12/2024 at 14:23, Maria sent: 'The address is Rua Augusta 42, 3rd floor, Lisbon.'

U Summarize my conversation with the landlord from last month.



Last month, you discussed 3 topics: 1) Rent payment date change (agreed to the 5th), 2) Plumbing issue (he'll send someone Friday), 3) Lease renewal (pending his response).

U Who sends the most messages in my group chat? Show me the statistics.



Participant stats: João (1,204 messages), Ana (892), Pedro (456). João sends 42% of all messages.

Frequently Asked Questions

01 How does the WhatsApp Chat Export Parser handle large chat files?

The system processes massive chats safely. It doesn't overload your agent by trying to read everything at once; instead, it intelligently sends only the first 100 and last 50 messages plus participant stats for analysis.

02 Is my WhatsApp chat data private when I use the Chat Export Parser MCP?

Yes. This service is designed to be completely air-gapped; all parsing happens locally on your device. Your private conversations never leave your machine or get uploaded to a cloud server.

03 Can I find an address from old messages using the Chat Export Parser MCP?

Absolutely. You can ask your agent to search for specific details, like addresses or phone numbers, and it will pull out the exact message with the date and sender.

04 What if my chat uses European date formats? Will the Chat Export Parser handle it?

Yes. The parser handles both US format (1/15/23) and EU format (15/01/2023) automatically, so you don't have to worry about international formatting differences.

05 Does the Chat Export Parser only work on group chats?

No. You can use it on any chat export, whether it's a one-on-one conversation or a massive team group discussion.

06 Are my private chats sent to the cloud?

Never. The parsing is 100% local. Only the structured text representation is sent to the AI chat context during your session.

07 How do I export a WhatsApp chat?

Open the chat in WhatsApp, tap the three dots > More > Export Chat > Without Media. Save the .txt file to your computer.

08 What if the chat has 50,000 messages?

The engine uses a token-safe strategy: it sends only the first 100 and last 50 messages, plus full participant stats. Ask the AI to filter by sender or date for specific results.

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 "whatsapp-chat-export-parser": { "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

WhatsApp Chat Export Parser 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 WhatsApp Chat Export Parser. 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	July 2026
MCP Server	WhatsApp Chat Export Parser MCP
Server ID	019e390a-1d39-7089-a587-24f7c4079219
Platform	Vinkius Cloud for AI Agents
Endpoint	<code>https://edge.vinkius.com/{token}/mcp</code>

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-chat-export-parser.