MCP SERVER | NO CODE | CLOUD HOSTED

Regex High-Perf Parser MCP

Extract patterns from massive logs with 100% accuracy.

Regex High-Perf Parser MCP. Stop losing data when you extract patterns from massive text blocks. This tool runs pure V8 Regular Expressions against gigabytes of logs or transcripts, guaranteeing every single match is returned in a complete JSON array. It delivers deterministic extraction where other AI systems fail due to context limits.

A+ Quality Score 100/100

regex | v8-engine | high-performance | data-parsing | entity-extraction | log-analysis

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/regex-high-perf-parser-alternative

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.

Regex High-Perf Parser MCP

1 tools available
Cloud-hosted on Vinkius

When you're dealing with large log files—say, thousands of lines detailing network activity—you need perfect data fidelity. Standard LLMs often truncate results or drop records simply because the input text exceeds their context window. This MCP solves that problem by executing standard V8 Regular Expressions directly on a local runtime environment. You feed it your massive text block and your specific pattern (the regex). The system processes everything, returning an exact, complete JSON array of every match found. There are zero dropped entities and no hallucinations. Because this tool runs outside the typical LLM context limits, you get reliable results whether you're hunting for IPv4 addresses or obscure order IDs. Vinkius hosts this MCP, giving your agent access to specialized data processing tools that go beyond general language capabilities.

Core Capabilities

01 — Extract exact patterns from text

You provide a block of text and a pattern, and the tool returns a guaranteed array containing every specific string match found.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/regex-high-perf-parser-alternative** — connect your AI agent in three steps.

01 Submit your massive source text (e.g., log file content) to the MCP.

02 Define your extraction rule using a standard Regular Expression pattern.

03 The tool runs the V8 engine, processes all data deterministically, and returns a complete JSON array of every single match.

The bottom line is you get flawless data extraction without hitting context limits or fighting model hallucinations.

Built For

This MCP is critical for DevOps engineers and data analysts who spend too much time manually checking log files or debugging unreliable AI extractions. If your job involves pattern matching on large, unstructured text at scale, you need this.

DevOps Engineer

Debugging network logs or application traces by extracting every unique IP address or error code from multi-gigabyte files.

Data Analyst

Processing customer support transcripts to extract all specific order IDs, user names, and dates mentioned in the chat history.

Security Engineer

Scanning massive server access logs to guarantee that every single suspicious IP address or malicious signature is captured for review.

What Changes When You Connect

01 Guaranteed Data Fidelity: Don't risk losing results. This MCP runs V8 Regex directly, ensuring every single match is found in the text block.

02 Handles Massive Scale: Context limits won't trip you up. Process huge log dumps or transcripts that would cause other AI clients to fail or truncate.

03 Predictable JSON Output: You get a clean, structured JSON array back. No messy, unstructured text dumps—just perfectly formatted results.

04 Universal Pattern Matching: Use standard regex rules for anything from IPv4 addresses and email formats to custom order IDs.

05 Fast Execution: The V8 engine is optimized for speed, making large-scale data parsing quick enough for real-time workflow needs.

Real-World Applications

Analyzing a massive web server log dump

A DevOps engineer needs to find every unique IP address that hit the API yesterday. They ask their agent to use the regex parser with an IPv4 pattern on the 10,000-line file. The MCP returns a complete list of all IPs in a reliable JSON array.

Extracting identifiers from complex transcripts

A data analyst is reviewing customer support chats and needs to pull every single order ID that matches the pattern `ORD-[A-Z0-9]{8}`. They feed the transcript into the regex parser, which guarantees all IDs are captured correctly.

Bulk email harvesting from internal documents

You have a large corpus of mixed text and need every email address extracted. You run the regex parser with an email pattern; the resulting structured JSON array gives you a perfect list for validation or bulk mailing.

Cleaning up messy data streams

A security team needs to check if any log entry contains specific compliance keywords. Instead of relying on fuzzy AI matching, they use the regex parser to strictly look for patterns like 'failed login' and get a list of every instance.

Patterns to Avoid

Assuming LLMs handle large logs

✗ AVOID

The engineer pastes 50,000 lines of log data into the chat prompt and asks, 'Find all IP addresses.' The AI client responds with a partial list and says it's done.

✓ INSTEAD

Use the regex_parser_extract tool. You pass the full log text and the specific IPv4 pattern to guarantee every single match is returned in one reliable JSON array.

Trying to find complex patterns in multiple steps

✗ AVOID

The analyst tries to extract IDs by asking the AI to 'find anything that looks like an order number, then list them.' The results are inconsistent and require manual cleanup.

✓ INSTEAD

Instead, define the exact pattern using regex_parser_extract. Passing the source text and the specific ID pattern ensures deterministic capture every time.

Using general search instead of structured extraction

✗ AVOID

The user pastes a big chunk of data and asks for 'all emails.' The model might miss emails embedded in complex sentences or fail if the text is too long.

✓ INSTEAD

Always use regex_parser_extract. It treats the task like pure computation, making sure that every string matching the email pattern gets captured regardless of context size.

The Right Fit

Use this MCP when your primary requirement is 100% deterministic data extraction based on a known pattern. If you are counting records or need to extract specific identifiers (like IP addresses, UUIDs, or custom IDs) from massive text blocks, this tool is essential. Don't use it if you need the AI to interpret meaning—if you ask it 'What happened here?' that requires LLM context. However, if the task involves purely pattern matching and array extraction across huge volumes of data, this MCP beats any general-purpose AI client because it bypasses context window limitations entirely.

Parsing logs means endless copy/pasting and hoping nothing gets missed.

Today, when a server throws a massive log dump or you're reviewing hundreds of support transcripts, the process is manual hell. You copy the block of text, paste it into your agent, run a prompt like 'Find all IPs,' and then you spend time checking if the model dropped any results because the context window was too full. It’s frustrating guesswork.

With this MCP, that process vanishes. You hand over the source text and the precise regex pattern once. The tool executes the extraction purely on V8, returning a mathematically exact JSON array of every single match. You get reliable data instantly, without needing to worry about context limits or hallucinated gaps.

Regex High-Perf Parser MCP Guarantees Perfect Pattern Extraction

You never have to copy a huge log file snippet and worry if the AI agent will choke on it. You don't have to manually check the results for missing IDs or dropped IP addresses.

What changes now is that data extraction becomes a reliable computation, not an interpretation. It’s fast, deterministic, and accurate.

Regex High-Perf Parser: 1 Tool

Use this single tool to take any text and a specific regex pattern, then reliably extract every matching piece of data into a structured JSON array.

#	TOOL	DESCRIPTION
01	`regex_parser_extract`	Accepts a source text and a regex pattern, returning an accurate array of all string matches found within the text.

See It in Action

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

U Use the regex parser with the pattern `\b\d{1,3}(\.\d{1,3}){3}\b` to extract every single IPv4 address from this massive server log.

The computation has been executed with mathematical precision. All results are exact and ready for review.

U Find all email addresses in this text block using regex and return them as a strict JSON array.

The computation has been executed with mathematical precision. All results are exact and ready for review.

U Extract all order IDs matching the pattern `ORD-[A-Z0-9]{8}` from this customer support transcript.

The computation has been executed with mathematical precision. All results are exact and ready for review.

Frequently Asked Questions

01 **How does Regex High-Perf Parser MCP handle context limits?**

The tool runs standard V8 Regular Expressions on a local runtime, completely bypassing the context window limitations that affect general AI client prompts. It processes massive inputs reliably.

02 **Can regex_parser_extract find every IPv4 address in a large log file?**

Yes. You provide the text and the IPv4 pattern, and the tool is designed specifically to guarantee that it captures every single instance of the matching pattern.

03 **Is this better than asking my AI client to extract data?**

For pure extraction tasks—like finding emails or order IDs—yes. This MCP runs computation, not interpretation, making its results deterministic and reliable at scale where LLMs tend to fail.

04 **What kind of patterns can I use with regex_parser_extract?**

You can use any standard V8 Regular Expression pattern. This covers everything from simple character groups to complex, nested patterns for identifiers.

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 `"regex-high-perf-parser-alternative": { "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

Regex High-Perf 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 Regex High-Perf 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	June 2026
MCP Server	Regex High-Perf Parser MCP
Server ID	`019eb8f6-bd46-72d8-9673-bf4a3da2dd5c`
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/regex-high-perf-parser-alternative.