

MCP SERVER

NO CODE

CLOUD HOSTED

Tempo Tap Analyzer MCP

Quantify rhythm stability from raw timing data.

Tempo Tap Analyzer quantifies rhythmic timing, BPM, and groove stability from raw tap sequences. This MCP lets you analyze any set of millisecond timestamps to derive precise musical metrics. You can assess whether a performance sounds human or robotic, determine underlying patterns like ternary rhythms, and calculate average beats per minute instantly. Stop guessing about rhythm; start measuring it.

A+ Quality Score 100/100

tempo

bpm

rhythm

groove

musicology

timing



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.

Tempo Tap Analyzer MCP

3 tools available

Cloud-hosted on Vinkius

This MCP gives musicians and musicologists the data they need to quantify groove and analyze rhythm structure. If you have raw timing information—a list of millisecond taps from a performance or sequence—this connector turns that messy data into actionable musical metrics. You can use your agent to assess the precision of any rhythm, determining if it has a steady, machine-like feel or natural human variation. Need to know what kind of beat structure is at play? It detects complex patterns like binary and ternary subdivisions. Plus, you can calculate average BPM and interval details from those tap timestamps.

When working with raw performance data, having instant access to this level of detail changes everything. Instead of running multiple calculations in a spreadsheet or guessing the pattern by ear, your agent handles it all. You connect once through Vinkius and gain immediate access to these specialized audio analysis tools.

Core Capabilities

01 — Determine overall rhythm speed

Calculate average BPM and precise interval details from a sequence of tap timestamps.

02 — Assess performance precision

Analyze the stability of a rhythmic sequence to judge if it sounds highly mechanical or naturally variable.

03 — Identify underlying beat patterns

Detect specific rhythmic structures, such as Binary, Ternary, or Quaternary subdivisions, present in the taps.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/mcp/tempo-tap-analyzer — connect your AI agent in three steps.

- 01 Provide your agent with a series of millisecond timestamps representing the tap sequence.
- 02 Select the analysis mode—whether you want to measure tempo, check stability, or find subdivisions.
- 03 Your MCP returns structured data including average BPM, interval breakdowns, and pattern classifications.

The bottom line is that you feed it raw timing numbers and get back musically structured metrics.

Built For

This is for the musicologist who analyzes historical recordings; the audio engineer needing to quantify performance quality; or the composer building complex rhythmic modules. If your work involves turning time into measurable musical data, this MCP saves hours of tedious calculation.

Musicologist

They analyze historical recordings by inputting tap data and using the MCP to identify specific subdivisions or measure how stable a rhythm was in a given era.

Audio Engineer

They use this tool to check recorded performances. By running stability analysis, they can determine if a musician's timing is too erratic or perfectly robotic for the intended sound.

Composer / Sound Designer

They test rhythmic ideas by calculating tempo metrics on raw tap data to ensure their modules hit exact BPM targets and maintain consistent intervals.

What Changes When You Connect

- 01 Stop relying on ear training. Use `calculate_tempo_metrics` to get precise average BPM and interval details, turning subjective listening into hard numbers.

-
- 02** Assess human performance quality instantly. `analyze_timing_stability` tells you if a track sounds highly natural or unnervingly mechanical, letting you fix timing issues before mixing.
-
- 03** Decode complex beats without guessing. `identify_rhythmic_subdivision` detects underlying structures like Ternary or Quaternary rhythms, giving context to your tap data.
-
- 04** Streamline analysis for music theory. You can run multiple tests on the same sequence—calculating tempo while simultaneously identifying subdivisions—all in one chat session.
-
- 05** Focus on creation, not calculation. By offloading the math (BPM, intervals, patterns) to this MCP, you spend less time troubleshooting spreadsheets and more time making music.
-

Real-World Applications

Evaluating a live performance recording

An audio engineer records a drum solo. Instead of manually reviewing the waveform for consistency, they ask their agent to run `analyze_timing_stability` on the tap data. The MCP confirms the timing is highly stable but suggests slight deviations would give it more 'human feel' for mixing.

Designing a beat-matching game module

A sound designer needs to hit exactly 128 BPM. They use `calculate_tempo_metrics` on a sample rhythm and confirm the average interval is precisely 500ms, guaranteeing perfect timing for their synth patch.

Analyzing ethnic drumming patterns

A musicologist has raw tap data from a specific cultural rhythm. They use `identify_rhythmic_subdivision` to confirm if the pattern is genuinely Ternary, providing concrete evidence for their academic paper.

Comparing rhythmic complexity

A composer inputs two different tap sequences into the MCP. They run both through `identify_rhythmic_subdivision` and see one is Binary while the other is Quaternary, helping them structure a more complex piece.

Patterns to Avoid

Treating timing analysis as basic math.

X AVOID

Just calculating BPM based on two endpoints (start and end time) gives you an inaccurate average because it ignores the rhythm's internal variations and intervals.

✓ INSTEAD

You must use `calculate_tempo_metrics`, which processes every tap timestamp to return a true average BPM *and* detailed interval information. Don't just look at the start/end points; look at the whole sequence.

Assuming all rhythms are simple.

X AVOID

If you manually check a rhythm and assume it must be Quarter notes, you might miss complex patterns like mixed Binary or Ternary subdivisions that require dedicated analysis.

✓ INSTEAD

Always use `identify_rhythmic_subdivision`. It checks for multiple underlying patterns, so you don't have to guess the structure of the groove.

Ignoring performance nuance.

X AVOID

A piece might sound 'off' just because it lacks natural variability. Simply calculating BPM won't tell you *why* it sounds unnatural.

✓ INSTEAD

You need `analyze_timing_stability`. This tool specifically grades the rhythm against a standard, telling you if the deviation is too consistent (robotic) or too chaotic.

The Right Fit

Use this MCP if your goal is to translate raw millisecond tap data into quantifiable musical metrics like BPM, interval detail, and pattern identity. This tool excels when the problem is 'What *is* the mathematical structure of this rhythm?' Don't use it if you just need to check general audio levels or filter noise; those are tasks for signal processing tools. Also, don't try to fix a musical composition using only this MCP; it diagnoses and measures, but doesn't generate audio or write sheet music. If your goal is purely creative generation, you need a different type of tool entirely. You use this when the data input (tap times) is perfect, but the analysis output needs rigor.

The Struggle with Rhythm Analysis

Right now, analyzing rhythm means dealing with spreadsheets full of timestamps and trying to eyeball patterns. You copy-paste raw tap data into a spreadsheet, manually calculating intervals between points, then you spend hours in audio software trying to find the underlying beat structure just by ear. If you want BPM, you run one formula; if you want subdivision? You start guessing.

With this MCP, that process disappears. You feed your agent the raw taps and get immediate, structured data back. It doesn't just give you a number; it tells you *why* that rhythm sounds the way it does, providing concrete metrics on stability and pattern type.

Tempo Tap Analyzer MCP: Structured Data When You Need It

You no longer have to juggle three different analyses. Instead of running one tool for BPM, another for patterns, and a third for general stability, you run them all sequentially through your agent. The system pulls the required metrics—average BPM from `calculate_tempo_metrics`, pattern names from `identify_rhythmic_subdivision`, and precision scores from `analyze_timing_stability`—and gives you one cohesive report.

What's different now is that analysis isn't a process of manual calculation; it's a single, direct data query. You ask the question about rhythm, and the MCP delivers the objective answer.

Tempo Tap Analyzer: 3 Tools Available

Use these three specialized tools to break down rhythmic sequences by measuring tempo metrics, assessing timing stability, and identifying underlying subdivisions.

#	TOOL	DESCRIPTION
01	<code>analyze_timing_stability</code>	Checks a given rhythm sequence to see how stable the timing is overall.
02	<code>identify_rhythmic_subdivision</code>	Finds and names the repeating rhythmic pattern or subdivision in your tap data (like Ternary or Binary).
03	<code>calculate_tempo_metrics</code>	Takes tap timestamps and calculates the average BPM, along with detailed interval measurements.

See It in Action

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

U Calculate the BPM for these taps: [0, 500, 1000, 1500]



The average BPM is 120. The intervals are all 500ms.

U How stable is this rhythm: [480, 520, 490, 510]?



The timing stability is high, indicating a very precise and robotic rhythm.

U What subdivision is present in these intervals: [500, 750, 500, 750]?



The detected subdivision is Ternary, with a high confidence score.

Frequently Asked Questions

01 How does Tempo Tap Analyzer calculate BPM?

It calculates BPM by using `calculate_tempo_metrics` on your tap timestamps. It doesn't just use the start and end times; it processes every interval between taps to determine a true average beat rate.

02 Can I tell if a rhythm is human or machine-made with this MCP?

Yes, you can. Running `analyze_timing_stability` assesses the sequence for precision. High stability suggests robotic timing, while natural variation indicates a more organic performance.

03 What kinds of patterns can identify_rhythmic_subdivision find?

It detects common rhythmic structures like Binary, Ternary, and Quaternary subdivisions. This helps categorize the underlying beat pattern in your data.

04 Does Tempo Tap Analyzer need specific file formats?

No, it requires raw millisecond timestamps as input. You'll feed these numerical values directly into the MCP through your agent.

05 Is this for professional musicology research?







Absolutely. The level of detail in interval measurements and pattern identification makes it suitable for deep academic analysis of musical performance data.

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 <code>"tempo-tap-analyzer": { "url": "..." }</code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → 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

Tempo Tap Analyzer 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 · 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 Tempo Tap Analyzer. 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	Tempo Tap Analyzer MCP
Server ID	019eff93-a3ed-722c-8d1a-9bbd509f9a0f
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/tempo-tap-analyzer.