MCP SERVER | NO CODE | CLOUD HOSTED

Roman Numeral Engine MCP for AI Agents

Convert and validate large-scale Roman numerals for historical and educational projects.

Roman Numeral Engine MCP converts numbers between Arabic integers and Roman numerals. It handles standard values and large-scale numbers up to 3,999,999 using vinculum notation. Use it to generate, decode, or audit Roman numerals accurately for historical projects, educational content, or specialized software.

A+ Quality Score 100/100

roman-numerals | arabic-to-roman | vinculum-notation | number-conversion | validation

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JULY 2026
vinkius.com/mcp/roman-numeral-engine

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.

Roman Numeral Engine MCP

3 tools available
Cloud-hosted on Vinkius

When you're working on a historical project or building a game that requires accurate Roman numerals, things get messy quickly once you go past 3,999. Most standard converters fail there, but this MCP handles the heavy lifting by using vinculum notation for those larger values. It's built for precision, whether you're trying to turn a massive integer into a readable string or checking if a specific notation follows strict historical rules. Instead of hunting for a reliable converter online or writing custom logic to handle overlines, you can just ask your agent to do it for you. It's one of the specialized tools you'll find in the Vinkius catalog that makes these specific formatting tasks disappear. You get consistent results every time, ensuring your data stays accurate across different contexts without the headache of manual calculation. This makes it a solid choice for anyone who needs to maintain high standards of accuracy in their documentation or UI.

Core Capabilities

01 — Convert integers to Roman numerals

Turn a standard number into a Roman numeral string instantly.

02 — Decode Roman numerals back to integers

Change a Roman numeral string back into a standard Arabic number.

03 — Check notation rules

Audit a Roman numeral string to see if it follows specific standards.

04 — Handle large values

Process numbers over 4,000 using correct vinculum overline notation.

05 — Audit historical accuracy

Verify if a string meets strict or historical notation requirements.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/roman-numeral-engine** — connect your AI agent in three steps.

01 Provide your agent with a number or a Roman numeral string.

02 The MCP processes the input based on the specific conversion rules.

03 You get back the converted value or a validation status.

The bottom line is you get accurate Roman numeral conversions for any size integer without manual calculation.

Built For

This is for historians, educators, and game developers who need precise number formatting for historical dates, titles, or UI elements. It solves the pain of manual conversion errors and the limitations of basic converters.

Historian

Validating dates and titles in archival research and publications.

Game Developer

Creating dynamic UI or quest markers for a Roman-themed game.

Educator

Generating practice problems and checking student work for history classes.

What Changes When You Connect

01 Handle massive values up to 3,999,999 using `convert_arabic_to_roman`.

02 Audit strings for accuracy using `validate_roman_notation`.

03 Decode complex strings back to integers with `convert_roman_to_arabic`.

04 Eliminate manual calculation errors for historical dates.

05 Choose between STRICT and HISTORICAL standards for specific project needs.

Real-World Applications

Historical Archive Research

A researcher needs to convert a large year like 4500 into a Roman numeral with an overline. They ask their agent to do it, and the MCP handles the vinculum notation correctly.

Game Development UI

A developer wants to generate a list of Roman numerals for a gladiator themed game UI. The agent uses the engine to create a consistent list of labels.

Educational Content Creation

A teacher wants to verify if a student-written list of Roman numerals is correct. The agent audits the strings for historical accuracy.

Data Formatting Projects

A user wants to convert a list of modern integers into Roman numerals for a specific aesthetic design. The agent processes the list in one go.

Patterns to Avoid

Trying to convert 5,000 without vinculum

✗ AVOID

Asking the agent to convert 5,000 and getting a standard string that ignores the overline rule.

✓ INSTEAD

Use `convert_arabic_to_roman` which is specifically built to handle large values using the correct overline notation.

Assuming all numerals are correct

✗ AVOID

Using a string like 'IIII' in a project that requires standard notation.

✓ INSTEAD

Use `validate_roman_notation` in STRICT mode to catch these errors before they go live.

Manual calculation

✗ AVOID

Trying to do the math in your head to figure out the correct symbols for 3,999.

✓ INSTEAD

Let the MCP handle the logic to ensure you never make a manual entry mistake.

The Right Fit

Use this if you need to handle numbers above 3,999 or require strict validation of historical notation. Don't use it if you only need basic 1-3,999 conversions and don't care about historical accuracy, as a simpler converter might suffice.

Roman Numeral Engine for Precise Historical Date Conversion

People often struggle with Roman numerals when the numbers get big. You find yourself manually trying to figure out where the overlines go or if 'IIII' is actually acceptable for your specific project. It's a tedious process of cross-referencing rules and doing mental math.

This MCP handles the logic for you. Just give your agent a number and it handles the vinculum notation and standard rules instantly. You get a perfect string every time.

Roman Numeral Engine for Automated Educational Validation

Teachers and content creators often have to check dozens of Roman numeral strings for accuracy. Manually checking every single one for historical consistency is a massive time sink.

You can now batch-validate these strings using the validation tools. It turns a manual audit into a quick check, making it much easier to grade assignments or verify content.

Roman Numeral Engine 3-Tool Historical Conversion

Convert integers to Roman numerals, decode strings, and validate notation for large-scale values.

#	TOOL	DESCRIPTION
01	`convert_arabic_to_roman`	Turns a standard number into a Roman numeral string.
02	`convert_roman_to_arabic`	Changes a Roman numeral string back into a standard Arabic number.
03	`validate_roman_notation`	Checks if a Roman numeral string follows strict or historical rules.

See It in Action

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

U Convert the number 5,500 to a Roman numeral.

The Roman numeral for 5,500 is $overline{ ext{V}}$D.

U Is 'XCIX' a valid Roman numeral?

Yes, **XCIX** is a valid Roman numeral.

U Convert 1,200 to a Roman numeral.

The Roman numeral for 1,200 is **MCC**.

Frequently Asked Questions

01 **Does the Roman Numeral Engine MCP support numbers over 4,000?**

Yes, it uses vinculum notation to handle values up to 3,999,999.

02 **Can I check if a Roman numeral is historically accurate?**

Yes, you can use the validation tool to check against strict or historical standards.

03 **What is vinculum notation?**

It's a system using an overline to represent thousands, allowing the engine to handle very large numbers.

04 **How does the Roman Numeral Engine MCP handle large numbers?**

It automatically applies the overline notation for any integer 4,000 or higher.

05 **Can I convert a Roman numeral back into a standard number?**

Yes, the engine can decode any Roman numeral string into its Arabic integer equivalent.

06 **Is there a way to check for 'IIII' vs 'IV'?**

Yes, the validation tool has a strict mode to ensure standard notation is followed.

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 `"roman-numeral-engine": { "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

Roman Numeral Engine 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

DOCUMENT INFORMATION

Generated	July 2026
MCP Server	Roman Numeral Engine MCP
Server ID	`019f3109-6fcd-71a9-8444-b26b99656c2c`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE