MCP SERVER | NO CODE | CLOUD HOSTED

RPG Damage Engine MCP for AI Agents

Calculating Complex Combat Rolls and Encounter Probabilities

The RPG Damage Engine calculates complex damage for tabletop role-playing games. It lets you parse advanced combat formulas, like '2d6+3 fire,' to instantly get the minimum, maximum, average, and standard deviation of dice rolls. You can also predict success rates against tough enemies by simulating multi-attack scenarios.

A+ Quality Score 100/100

ttrpg | dnd | pf2e | dice-rolling | probability | damage-calculation

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JULY 2026
vinkius.com/mcp/rpg-damage-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.

RPG Damage Engine MCP

4 tools available
Cloud-hosted on Vinkius

Running a TTRPG session means keeping track of math: calculating damage from mixed sources, applying resistances, or predicting if the party has a shot at taking down that heavily armored boss. This MCP handles all of that complexity. You feed it a raw combat expression—say, '1d8+4 piercing'—and it immediately breaks down the stats you need, including the average expected damage and the range of outcomes. Beyond simple rolls, it supports advanced mechanics found in systems like D&D 5e or PF2e, letting you apply advantage/disadvantage modifiers or account for elemental vulnerabilities. It even lets you predict overall encounter difficulty using complex probability modeling. When your AI client is connected through Vinkius, you stop calculating things by hand and start focusing on the story. Your agent takes over the math so you can keep the dice rolling.

Core Capabilities

01 — Analyze damage formulas

It breaks down complex combat expressions into their core components for immediate statistical review.

02 — Calculate encounter probability

Predicts the chance of defeating an enemy by simulating multiple attack distributions.

03 — Evaluate modified damage

Determines the final damage number after applying specific rules like resistances, vulnerabilities, or critical hits.

04 — Parse raw combat expressions

Deconstructs a messy damage string into usable numbers and types for calculation.

One Click on Vinkius — From Prompt to Execution

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

01 You give your AI client the raw damage expression or scenario you need analyzed, such as '2d6+3 fire' or an opponent's HP pool.

02 The MCP processes this input, running it through its specialized combat math engine to model all possible outcomes and statistical parameters.

03 Your agent returns a clean report detailing minimum damage, maximum damage, average expected value, and probability metrics.

The bottom line is you get precise, system-accurate combat math without opening three different spreadsheets.

Built For

This MCP is built for anyone who runs or designs tabletop RPGs. If tracking complex damage rolls and predicting outcomes feels like a chore at the end of a long session, this tool saves your sanity.

Game Master (GM)

Needs to quickly calculate enemy damage output or predict if a party has enough resources to survive a multi-enemy encounter.

TTRPG Player

Wants a reliable way to double-check the math on complex rolls, especially when dealing with specialized rules like advantage/disadvantage.

Game Designer

Requires rigorous statistical analysis of new combat mechanics or damage scaling to ensure game balance across multiple levels.

What Changes When You Connect

01 Stop guessing. Use calculate_encounter_probability to predict your success rate against high-HP foes, so you never roll blind.

02 Understand every number using analyze_damage_expression. It gives you the minimum, maximum, average, and standard deviation for any damage formula.

03 Model advanced rules easily with evaluate_modified_damage. It correctly adjusts damage when applying things like elemental resistances or critical hit multipliers.

04 Save time parsing raw text. parse_damage_expression breaks down messy combat strings into usable components instantly.

05 Keep the game flowing. By handling the math, you spend less time on calculation and more time describing epic battles.

Real-World Applications

The Boss Fight Math Problem

A GM needs to know if a group of three fighters can reliably take down a boss with 150 HP. They ask their agent, which uses calculate_encounter_probability, and get a clear probability score, letting them adjust the difficulty on the fly.

Testing New Spell Damage

A designer writes a new fire spell formula: '3d6+2 fire'. They use analyze_damage_expression to confirm its average damage is 20.5, and it has a low standard deviation, meaning the output is predictable.

Resistances and Vulnerabilities

A player hits an enemy with '2d8+4 acid' but the monster resists acid (0.5 multiplier). They ask their agent to use evaluate_modified_damage, getting the exact adjusted average damage: 13.

Understanding Complex Inputs

The GM finds a note with '2d4+3 piercing.' Instead of guessing, they run parse_damage_expression to break it down into its distinct components (dice count, dice type, modifier) before calculating anything.

Patterns to Avoid

Treating damage as a single number

✗ AVOID

Assuming that '2d6+3' means you will always hit for 15. This ignores the huge variance and potential low rolls, leading to unbalanced combat.

✓ INSTEAD

Always use analyze_damage_expression. It shows you Min (4), Max (14), and Average (11). Never rely on a single number.

Forgetting modifiers

✗ AVOID

Calculating base damage for an elemental attack without factoring in the target's resistance, over-inflating the threat level.

✓ INSTEAD

Use evaluate_modified_damage. It accounts for resistances and vulnerabilities automatically, giving you the true final number.

Ignoring overall probability

✗ AVOID

Just calculating damage per attack but failing to account for multiple enemy attacks or multi-stage encounters.

✓ INSTEAD

Run calculate_encounter_probability. It models the entire fight, giving you a single prediction of success chance.

The Right Fit

Use this MCP if your game relies on strict mathematical rules and complex probability modeling. Specifically, when you need to predict outcomes (like calculating_encounter_probability) or adjust base damage with modifiers (evaluate_modified_damage). Don't use it if you just need basic dice rolls; a simple random number generator is fine for that. You should avoid this MCP if your game system uses vague descriptors instead of quantifiable formulas. If the math is messy but predictable, this MCP handles it well. However, remember that while tools like parse_damage_expression help read the inputs, they don't make up rules; you still need to provide the full context for accurate results.

RPG Damage Engine: Mastering Complex TTRPG Combat Math

Right now, if a GM needs to calculate damage involving multiple dice types or combat modifiers (like advantage), they're doing it by hand. That means shuffling notes, cross-referencing system rulebooks, and constantly running the math in separate calculators. It’s slow, prone to human error, and totally breaks immersion.

With this MCP, your agent handles all that complexity for you. You give it a scenario—like '2d6+3 fire' with advantage—and instantly get the full statistical picture: Min, Max, Average, and Standard Deviation. It’s combat math solved.

RPG Damage Engine: Predicting Encounter Difficulty in Gaming

Before this MCP, estimating how hard a boss fight would be meant guessing. You'd calculate damage per round for each character and try to manually estimate the enemy’s survivability against multiple attacks.

Now you can simply run predict encounter probability. It simulates all multi-attack distributions to give you a clear percentage of success or failure, letting you balance your session before the dice even touch the table.

RPG Damage Engine: 4 Tools for TTRPG Combat Math

Use these tools to analyze formulas, predict encounter outcomes, and calculate final damage numbers with precision.

#	TOOL	DESCRIPTION
01	`calculate_encounter_probability`	Predicts the likelihood of successfully defeating an enemy given their stats and your party's attacks.
02	`analyze_damage_expression`	Provides a full statistical breakdown (min, max, average, std dev) for any complex damage formula.
03	`evaluate_modified_damage`	Calculates the final damage number after applying combat modifiers like resistances or critical rules.
04	`parse_damage_expression`	Takes a raw text description of damage and breaks it down into its numerical parts for analysis.

See It in Action

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

U Analyze the damage for '2d8+5 piercing' with disadvantage.

Damage Analysis Report

- **Base Formula:** 2d8 + 5
- **Modifier Applied:** Disadvantage (0.7 multiplier)
- **Minimum Damage:** 3.5
- **Maximum Damage:** 14
- **Average Expected Damage:** 9.8

The average damage is significantly reduced, making this a high-risk, low-reward attack.

U What's the chance of dealing at least 30 total damage across 4 attacks using '1d10+2'?

Encounter Probability Simulation

Across 4 attacks, the probability of exceeding a total damage threshold of 30 is approximately **0.18** (or 18%).

Average Total Damage:* 24.
Recommended Action:* Consider increasing modifiers or reducing enemy HP to raise this chance.

U Analyze the damage for a critical hit: '3d6+5 fire' with resistance.

Modified Combat Evaluation

STATISTIC	VALUE
Base Damage (Crit)	3d6 + 5
Resistance Applied	x 0.5
Final Average Damage	**12.5**

The resistance significantly cuts the expected damage, so plan for a less lethal encounter.

Frequently Asked Questions

01 **How does the RPG Damage Engine calculate complex hit probabilities?**

It uses advanced simulation models to factor in multiple dice rolls and enemy defenses. You simply tell it the scenario, and it gives you a reliable percentage chance of success or failure for that encounter.

02 **Can I use the RPG Damage Engine to check damage with different elemental resistances?**

Yes. The engine handles modifiers like resistance and vulnerability automatically. You get the final, adjusted damage number you should expect after all system rules are applied.

03 **What kind of damage expressions can I feed into the RPG Damage Engine?**

It supports most standard TTRPG formulas, like '2d6+3 fire.' You just describe the formula in plain text, and it breaks down all the statistical properties for you.

04 **Is this MCP better than using a separate dice calculator?**

Yes. A basic calculator only gives you random rolls. This engine calculates *statistics*—the average, min, max, and probability—so you get actionable data for game balance, not just a single number.

05 **How does the RPG Damage Engine handle critical hits?**

It incorporates common critical hit rules from major TTRPG systems. You input the base damage, and it runs the calculations as if a crit occurred, giving you the right expected outcome.

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 `"rpg-damage-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

RPG Damage 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

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by RPG Damage Engine. 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	RPG Damage Engine MCP
Server ID	`019f2ba5-3a24-7279-b757-6d81cc7be95a`
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/rpg-damage-engine.