MCP SERVER | NO CODE | CLOUD HOSTED

# NIST NVD MCP

## Know every vulnerability affecting your product.

NIST NVD provides direct, conversational access to the National Vulnerability Database (NVD). Your agent can search for common vulnerabilities and exposures (CVEs), map threats to specific products using CPE strings, or analyze risk based on severity levels. It's your single source for authoritative cybersecurity product data.

A+ Quality Score 100/100

cve | cybersecurity | vulnerability-management | threat-intelligence | security-standards | product-security

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/nist-nvd

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.

# NIST NVD MCP

10 tools available
Cloud-hosted on Vinkius

Think of this MCP as a direct line to the global repository of security weaknesses. You connect it through Vinkius, giving your AI agent access to the world's most comprehensive archive of vulnerability and product information.

Instead of jumping between government sites or running complex queries in a dashboard, you just talk to your agent. Need to know if 'Microsoft Word 2019' has any known critical flaws? Ask it. Want to check every weakness associated with a specific component version? It handles that too. You can filter threats by how severe they are—Low, Medium, or Critical—to prioritize what needs fixing right now. The tool also lets you track changes in the database over time, so you always know if a threat was recently added or updated. This capability makes it an essential resource for anyone managing digital risk.

## Core Capabilities

**01 — Identify vulnerabilities by product**

You can find all known flaws linked to a specific piece of software or hardware using its official Common Platform Enumeration (CPE) string.

**02 — Search for weaknesses by type**

The MCP lets you look up vulnerabilities based on the underlying weakness, like CWE-89, rather than just knowing the CVE ID.

**03 — Filter threats by risk level**

You can narrow down thousands of results to see only those vulnerabilities rated as Critical or High severity for immediate action.

**04 — Track historical changes**

Retrieve a log detailing when vulnerability records were published, modified, or updated in the NVD database.

**05 — Query official product dictionaries**

You can search the CPE dictionary by simple keywords to identify potential software and hardware products involved in an exploit.

## One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/nist-nvd** — connect your AI agent in three steps.

01 Subscribe to this MCP via Vinkius. You might need your NIST NVD API Key if you expect high usage.

02 Direct your natural language query to your AI client, referencing the product or threat details needed.

03 The agent uses the relevant tool to search and return a structured list of vulnerabilities, CPEs, severity scores, or historical data.

The bottom line is you get authoritative vulnerability intelligence without writing complex API calls or navigating dense government websites.

## Built For

This MCP is for security analysts and DevOps engineers who are tired of manual threat hunting. If your job involves correlating known vulnerabilities to the software stack you manage, this tool saves hours of painful cross-referencing.

**Security Analyst**

They use this MCP to quickly gather CVE details or CVSS scores when assessing a new risk, helping them prioritize mitigation efforts immediately.

**DevOps Engineer**

They monitor the system for newly published vulnerabilities affecting specific software versions in their deployment pipeline.

**Compliance Officer**

They automate the gathering of vulnerability data required for security audits and generating compliance reports across multiple systems.

## What Changes When You Connect

01 Stop guessing about risk. Use search_cve_by_severity to filter thousands of results down to only Critical or High-risk threats, letting you focus on immediate patching needs.

02 Pinpoint affected products instantly. Running search_cve_by_cpe correlates vulnerabilities directly with a product's official CPE string, eliminating guesswork about scope.

03 Contextualize your findings. Instead of just seeing a CVE ID, get detailed information via get_cve_by_id and understand the full impact on your infrastructure.

04 Stay ahead of zero-days. Use search_cve_by_date to monitor only vulnerabilities published in the last 48 hours, ensuring you track emerging threats rapidly.

05 Validate product scope using get_cpe_by_id. If you aren't sure what the official CPE for a piece of software is, this tool gives you the authoritative reference needed before running any vulnerability checks.

## Real-World Applications

**Responding to an incident report**

A security analyst receives a suspicious alert mentioning 'Log4j' and needs immediate context. They ask their agent, which uses search_cve_by_keyword, to pull all relevant CVEs and then use search_cve_by_severity to filter the list down only to those rated Critical, providing an actionable remediation list.

**Onboarding a new product**

A DevOps engineer is deploying a new internal microservice. They ask their agent to search_cpe_by_keyword for all components used in the stack, then use search_cve_by_cpe on each component's CPE ID to guarantee no known flaws are present before launch.

**Preparing for an audit**

A compliance officer needs proof of due diligence regarding outdated software. They instruct their agent to search_cve_by_date for vulnerabilities published in the last quarter, and then use get_cve_change_history to prove they are tracking timely updates.

**Deep dive threat hunting**

A researcher needs to understand a specific type of weakness. They ask their agent to search_cve_by_cwe, targeting only injection flaws (CWE-89), and then use get_cve_by_id on the most severe results for technical details.

# Patterns to Avoid

### Treating NVD as a search engine

✗ AVOID

Asking your agent, 'What's wrong with my Windows PC?' This is too vague and yields useless results because the tool requires specific product identifiers or keywords.

✓ INSTEAD

Be precise. Instead of general questions, use get_cpe_by_id to find the exact CPE for 'Windows 11', then run search_cve_by_cpe with that identifier. This guarantees relevant and actionable results.

### Ignoring severity context

✗ AVOID

Pulling a massive list of CVEs from an entire product line, getting bogged down in thousands of 'Low' severity issues.

✓ INSTEAD

Always filter first. Use search_cve_by_severity to restrict results immediately to Critical or High. This focuses your attention on the vulnerabilities that pose the greatest risk.

### Manually cross-referencing data

✗ AVOID

Copying a CPE string from one spreadsheet, pasting it into a browser search, then manually checking multiple vulnerability databases.

✓ INSTEAD

Let your agent do it. Pass the CPE to search_cve_by_cpe and get an immediate, comprehensive list of all related CVEs in one go.

## The Right Fit

Use this MCP when you need authoritative data on *known* vulnerabilities tied to specific products or weaknesses. If your goal is gap analysis—figuring out what's missing from your code base—this is perfect because it gives you the reference data (CPE, CVE). However, don't use this if you need real-time runtime scanning of an active server for zero-day exploits; this MCP only accesses published historical records. If you are trying to compare vulnerability severity against internal risk scoring models, you'll want a specialized governance tool instead. Use it when the core question is: 'Is Product X vulnerable to Flaw Y?'

## Threat intelligence gathering used to be slow and fragmented.

Today, checking for vulnerabilities involves navigating multiple government sites or running complex command-line searches. You copy a product name into one search engine, then the CPE string into another, and finally cross-reference dates across third-party feeds. It's tedious work that often means missing critical information because you can't keep all those context windows open at once.

With this MCP, your agent handles the entire process conversationally. You just ask about a product or weakness. The system coordinates calls to find CPE matches, identify related CVEs, and sort them by severity—giving you a single, immediate risk assessment.

## get_cve_by_id: Direct access to critical vulnerability details

The biggest time-sink was having to manually look up the full technical description for every CVE ID you found. You'd copy a number, paste it into Google, and sift through pages of developer notes just to understand what kind of exploit it was.

Now, when your agent retrieves details using get_cve_by_id, you get the entire vulnerability profile—CVSS score, affected versions, and exploitation mechanism—in one clean response. It’s instant context for every single threat.

# NIST NVD: 10 Vulnerability Tools

These tools allow you to query the National Vulnerability Database for specific CVEs, map products via CPE identifiers, and filter threats based on severity or weakness type.

| # | TOOL | DESCRIPTION |
|---|---|---|
| 01 | `get_cpe_by_id` | Retrieves a specific entry from the CPE product dictionary using its unique UUID. |
| 02 | `get_cve_by_id` | Fetches all details for a known Common Vulnerabilities and Exposures (CVE) identifier. |
| 03 | `get_cve_change_history` | Retrieves a detailed log showing when a specific CVE entry was modified or updated in the NVD database. |
| 04 | `list_cpe_matches` | Lists valid CPE match strings to help you understand product scope and applicability. |
| 05 | `search_cpe_by_keyword` | Searches the entire product dictionary for entries matching a specified keyword. |
| 06 | `search_cve_by_cpe` | Finds all associated CVE vulnerabilities that affect a specific product defined by its CPE string. |
| 07 | `search_cve_by_cwe` | Searches for CVEs linked to a weakness type, such as CWE-89 (SQL Injection). |
| 08 | `search_cve_by_date` | Filters the database to find CVEs that were published or modified within a specific date range. |
| 09 | `search_cve_by_keyword` | Performs a broad search across all vulnerability data using general keywords. |
| 10 | `search_cve_by_severity` | Narrows down the results to only show CVEs that meet a specific CVSS severity level (e.g., Critical). |

# See It in Action

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

**U** Get the details for CVE-2023-23397.

Retrieving CVE-2023-23397... This is a Critical elevation of privilege vulnerability in Microsoft Outlook with a CVSS score of 9.8. It allows an attacker to steal NTLM hashes. Would you like to see the list of affected software versions (CPEs)?

**U** Search for vulnerabilities in 'WordPress' with CRITICAL severity.

Searching NVD... I've found several critical vulnerabilities affecting WordPress core and popular plugins. The most recent include CVE-2023-XXXX (Remote Code Execution). Shall I provide the full description for the most recent one?

**U** What is the official CPE name for 'Windows 11'?

Querying CPE dictionary... The primary CPE 2.3 name for Windows 11 is 'cpe:2.3:o:microsoft:windows_11:-:*:*:*:*:*:*:*'. I also found specific versions for various builds. Would you like to search for CVEs affecting this specific CPE?

# Frequently Asked Questions

01 **How do I find all vulnerabilities affecting 'Apache Struts' using NIST NVD?**

You can first search_cve_by_keyword with 'Apache Struts'. Then, use the CPEs found to run search_cve_by_cpe for a complete list of related CVEs.

02 **Can I check if my current software versions are listed in NIST NVD?**

Yes. You can start by using search_cpe_by_keyword to find the official CPE name, and then pass that identifier into search_cve_by_cpe to see all known flaws.

03 **Does NIST NVD help me prioritize which vulnerabilities to fix?**

Absolutely. Use search_cve_by_severity to filter results by CVSS score—you can narrow the focus instantly to Critical, High, or Medium risks for quick action.

04 **What is the difference between get_cve_by_id and search_cve_by_keyword?**

get_cve_by_id gives you everything about one specific flaw (e.g., CVE-2023-1234).
search_cve_by_keyword finds all flaws related to a general topic or component name.

05 **How do I know if the vulnerability data is recent?**

Use search_cve_by_date. This tool lets you narrow down results based on publication date, ensuring your assessment covers only recently reported threats.

# 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 `"nist-nvd": { "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

# NIST NVD 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 NIST NVD. 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 | NIST NVD MCP |
| Server ID | `019d75dd-f96b-71fb-b62f-6e3a67b1666a` |
| 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/nist-nvd.