

MCP SERVER

NO CODE

CLOUD HOSTED

Veracode MCP

Analyze code flaws, talk to your security findings.

Veracode connects your AI agent directly into your AppSec ecosystem. You stop clicking through dashboards and start asking questions about code security, vulnerabilities, and application risk profiles conversationally. Get a unified view of flaws across SAST, DAST, and SCA tools instantly.

A+ Quality Score 100/100

appsec

sast

dast

software-composition-analysis

code-security

devsecops



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.

Veracode MCP

10 tools available

Cloud-hosted on Vinkius

You can give your AI client deep read and write access to your Veracode environment, moving app security management out of the console and into natural conversation. Instead of logging in and hunting through multiple tabs—one for component analysis, one for dynamic scans, another for static flaws—you simply ask about a specific application or vulnerability type. For instance, you can request a summary listing all open findings across Static, Dynamic, and Component analytics right away. If you spot an issue, you don't just get a vague ID; the agent pulls up the underlying CWE error, affected code strings, and even remediation steps for you. This capability, now available through Vinkius, lets your AI act like a dedicated security engineer sitting next to you. You can manage entire application portfolios by creating new profiles or checking general health status without ever touching a settings menu.

Core Capabilities

01 — Manage Application Profiles

List all tracked applications, create new ones before code commits, and retrieve detailed profile information like business criticality and deployment state.

03 — Review Scan Statuses

List configured Dynamic Analysis scans or poll for real-time execution bounds of scheduled Web Application Security tests.

02 — Trace Specific Vulnerabilities

Retrieve a unified list of security flaws across an application and get deep details on specific findings, including the CWE error and necessary fixes.

04 — Identify Authorized Users

Audit the system by listing all users with Veracode access to manage roles and permissions.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/mcp/veracode — connect your AI agent in three steps.

- 01 Subscribe to this MCP and securely provide your dual Veracode API ID and API Secret pair.
- 02 Connect your preferred AI client (Claude, Cursor, Windsurf, etc.) to the Vinkius catalog.
- 03 Engage directly with your agent by querying security questions—for example, asking to list all apps or explain a specific finding.

The bottom line is that your AI agent translates complex security APIs into simple chat commands.

Built For

This MCP is for the DevSecOps engineer who's tired of clicking through dashboards at 2 am. It's for developers who need immediate, contextual flaw remediation and CISOs who must track enterprise risk across dozens of applications without manual effort.

DevSecOps Engineer

Uses the agent to check scan statuses or export flaws by chatting internally instead of opening multiple console tabs.

Application Developer

Commands the agent in their IDE to read a flawed line directly from a Veracode finding ID report, speeding up remediation time.

CISO & Security Manager

Audits all identity users or tracks general application risk matrices by reading human-summarized text outputs for compliance reports.

What Changes When You Connect

- 01 Get an immediate, unified view of all open security issues by asking the agent to list findings across SAST, DAST, and SCA tools. You skip opening ten different dashboards just to get a summary.

-
- 02 Drill down into flaws with `get_finding_details`. Instead of reading vague error codes, you immediately get an explanation of the underlying CWE error, affected code strings, and automated remediation steps.

 - 03 Manage your entire portfolio easily. Use `list_applications` and then `get_application_details` to check a project's risk score, business criticality rating, or compliance policy status in one chat session.

 - 04 Accelerate the development cycle by letting your agent read flaws directly from reports when you're coding. You can use this capability within Cursor or other IDEs.

 - 05 Keep an eye on environment readiness. The MCP lets you list sandboxes and poll for dynamic scan execution bounds, ensuring your testing environments are actually running what they should be.
-

Real-World Applications

Auditing App Risk Across the Board

A CISO needs to report on application risk before a board meeting. They ask their agent: 'List all applications and tell me which ones are marked as Very High business criticality.' The MCP uses `list_applications` followed by `get_application_details` for every result, giving them a single, summarized list of high-risk assets.

Reviewing App Deployment Readiness

A DevSecOps engineer needs to know if their new microservice is ready for production. They ask: 'What's the current security status of the Auth-Microservice?' The agent calls `list_security_findings` and summarizes any open issues across all three scan types.

Pinpointing the Root Cause of a Flaw

A developer sees an error ID and needs to fix it fast. They ask: 'What does finding ID 89 mean, and how do I patch it?' The agent calls `get_finding_details`, returning the CWE type, the exact file/line number, and a full remediation tutorial.

Managing User Access

A security manager needs to verify who has admin rights. They ask: 'Who are the authorized users in Veracode?' The agent calls `list_veracode_users` and presents a clean, readable list of all active accounts.

Patterns to Avoid

Treating it like a database query

X AVOID

Asking the agent: 'SELECT * FROM veracode.findings WHERE severity = critical AND app_id = GUID.' This forces rigid, unnatural syntax that isn't part of the chat flow.

✓ INSTEAD

Instead, just ask naturally: 'What are the critical flaws for my Mobile-Banking-iOS application?' The agent handles all the necessary function calls like ``list_security_findings`` and filters the results conversationally.

Assuming global context

X AVOID

Asking: 'Tell me about the latest vulnerability.' The agent has no idea which app or scan you mean, resulting in a generic error message.

✓ INSTEAD

Always provide context. Say: 'For the Legacy-CRM-Core application, what are the open findings?' This allows the agent to correctly scope the search using ``list_security_findings``.

Trying to manually manage profiles

X AVOID

Logging into the web UI just to check if a new profile exists or needs deletion. This is slow and involves multiple clicks.

✓ INSTEAD

Use the agent to list your applications with ``list_applications``, then use ``get_application_details`` for specific data, or even tell it to create one using ``create_application``.

The Right Fit

Use this MCP if your primary bottleneck is converting dense, multi-tab security reports into actionable text summaries. If you need an AI agent to read complex risk matrices, compare SAST findings against DAST results, or explain the technical remediation steps for a specific CWE flaw—this is the tool. Don't use it if you are trying to build a CI/CD pipeline that automatically executes scans; those systems handle execution flow better. Also, don't rely on this MCP for writing code fixes itself, though it can give you the patch instructions. This is about *reading* and *managing* the security data, not generating the entire fix from scratch.

The current way of tracking application risk feels like a scavenger hunt.

Right now, checking an app's security posture means opening Veracode. You click into the SAST report to see code flaws, then open the DAST tab for runtime issues, and maybe you jump to another section just for component analysis. You end up juggling three or four different dashboards, manually cross-referencing finding IDs and severity ratings across all of them.

With this MCP, you simply ask your agent: 'What are the top 5 critical flaws in our mobile banking app?' The agent pulls together the necessary data from every scan type—SAST, DAST, SCA—and presents a single, coherent summary. You get answers instantly without leaving your chat interface.

Veracode AppSec: Getting Clarity on Flaws with Veracode MCP

You stop manually exporting CSV reports and pasting them into spreadsheets just to get a high-level summary for management. You no longer have to click through menus just to see if an app passed its compliance check.

Now, you ask the agent: 'Give me the risk matrix for all apps.' It immediately runs checks like `list_applications` and provides summarized text outputs that your team can use right away. It's a fundamental shift from data retrieval to knowledge transfer.

Veracode: 10 Tools for AppSec Analysis

Use these tools to perform everything from listing all monitored applications to getting deep technical details on specific security vulnerabilities.

#	TOOL	DESCRIPTION
01	<code>create_application</code>	Creates a new Veracode profile container using the provided app schema and name.
02	<code>delete_application</code>	Permanently removes a specified application from the Veracode ecosystem. This action cannot be undone.
03	<code>get_api_health</code>	Checks the current operational status and connectivity health of your Veracode account.
04	<code>get_application_details</code>	Retrieves a detailed profile, including risk scores, business criticality rating, and policy compliance for one app.
05	<code>get_finding_details</code>	Gets precise technical details on a vulnerability, explaining the type, affected code file, path, and fix guidance.
06	<code>list_applications</code>	Returns a comprehensive list of all Veracode AppSec Applications currently tracked in your account.
07	<code>list_dynamic_analyses</code>	Lists all configured Dynamic Analysis (DAST) scans that are set up for your applications.
08	<code>list_security_findings</code>	Retrieves a unified summary of security findings across an entire application's monitored scope.
09	<code>list_sandboxes</code>	Lists every available testing sandbox that is linked to a specific application profile.
10	<code>list_veracode_users</code>	Shows the list of authorized identity users who have access and roles within Veracode.

See It in Action

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

U List all applications currently monitored in our Veracode account.



I've fetched 3 core tracked applications from your Veracode environment:

- 'Mobile-Banking-iOS' (GUID: 82a1...)
- 'Legacy-CRM-Core' (GUID: f3b9...)
- 'Auth-Microservice' (GUID: 11c2...)

Would you like me to pull the security finding logs specifically for 'Auth-Microservice'?

U Get the detailed security profile for the application GUID 'f3b9...!.



Here is the detailed profile for GUID `f3b9-12aa-99bb` : This corresponds to 'Legacy-CRM-Core'. It currently has a 'Very High' Business Criticality rating. Its compliance policy dictates adherence to 'Veracode Recommended High'. However, its recent automated scan status indicates 'Did Not Pass' due to 12 unmitigated critical severity flaws.

U Explain finding ID '89' from that app and how to fix it.



Finding ID '89' points to an Improper Neutralization of Input (CWE-79), better known as Stored Cross-Site Scripting (XSS). It was found in `user_profile.js` at line 144.

Description: Externally supplied, untrusted input is reflected directly into browser DOM executing scripts.

Remediation: To fix this, encapsulate the rendered variable `userInput` with a context-savvy encoding library like `DOMPurify` before injecting it via `innerHTML` .

I can write a quick patch for line 144 if you'd like.

Frequently Asked Questions

01 How do I list all the applications monitored in Veracode using the Veracode MCP?

You use the `list_applications` tool. This command pulls a full list of every AppSec Application currently tracked, giving you the GUIDs you need for further lookups.

02 Can I get detailed information about a specific finding using Veracode MCP?

Yes, use `get_finding_details`. You just give it the flaw ID, and the agent returns the vulnerability type (CWE), affected code, severity rating, and remediation guidance.

03 What is the difference between listing applications and getting application details with Veracode MCP?

`list_applications` gives you a simple list of names and GUIDs. `get_application_details` takes one of those GUIDs and returns deep metadata, like its business criticality rating or compliance policy.

04 If I want to delete an app profile, which tool do I use with the Veracode MCP?

You use `delete_application`. Be warned, this action is irreversible, so make sure you know what you're deleting before confirming.

05 How can I check if my API connection to Veracode works with the Veracode MCP?

Run `get_api_health`. This tool checks the current status of your connection and confirms that the necessary credentials are valid for use.

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 `"veracode": { "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

Veracode 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 Veracode. 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	Veracode MCP
Server ID	019d761b-6712-713c-b592-56c679da5615
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/veracode.