MCP SERVER | NO CODE | CLOUD HOSTED

HCL AppScan MCP

Audit entire app security posture instantly.

HCL AppScan MCP connects application security testing directly to your AI client. It lets you manage complex security scans across multiple applications, track vulnerabilities, and audit an entire software inventory using natural conversation. Quickly check scan statuses, list apps, or even start new dynamic analysis (DAST) tests without ever leaving your chat window.

A+ Quality Score 100/100

application-security | security-testing | vulnerability-management | code-auditing | threat-detection | devsecops

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

SCAN TO VIEW

The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect 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.

HCL AppScan MCP

10 tools available
Cloud-hosted on Vinkius

This MCP brings powerful application security testing straight to your agent. Instead of logging into separate dashboards, you can monitor vulnerabilities and audit your entire application inventory using natural conversation. Your AI client talks directly to the tools here, giving you instant insight into your security posture across HCL AppScan on Cloud (ASoC). You can list all applications in your inventory to find their unique IDs or check the real-time status of any active scan. Need more detail? You can retrieve detailed lists of security issues found during scans, including severity and current status. If you're ready for a new audit, you can start DAST scans right from the chat interface. All this capability is available through Vinkius, giving your agent access to industry-leading tools without needing multiple subscriptions or logins.

Core Capabilities

01 — Audit Application Inventory

You list all applications in your security inventory to get their unique IDs and names.

02 — Check Scan Statuses

You monitor all performed scans, checking the current status of any active security tests.

03 — Identify Vulnerabilities

You get detailed lists and specific information about security issues found during a scan.

04 — Initiate Security Scans

You start new Dynamic Analysis (DAST) scans for your web applications directly from the chat.

05 — Manage Internal Agents

You list available local agents used to scan internal, non-web applications.

One Click on Vinkius — From Prompt to Execution

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

01 First, your AI client uses the account tools to verify connection and retrieve basic user data.

02 Next, you ask it to list all applications or check a specific scan's status. The MCP runs those checks and sends back structured data about the findings.

03 Finally, if you need new data, you tell your agent to start a DAST scan; it executes the request and confirms when the job begins.

The bottom line is that your AI client handles the complex API calls so you just talk to it like normal.

Built For

Security Engineers, DevSecOps Teams, and Compliance Officers. This MCP helps people who get burned out logging into five different dashboards just to check if an app is compliant or secure.

Security Engineer

You audit security findings across multiple apps in one chat session, skipping the manual process of exporting data from separate consoles.

DevSecOps Team Lead

You integrate vulnerability tracking and scan initiation into automated developer workflows without writing boilerplate code to manage API calls.

Compliance Officer

You monitor application security status across the entire portfolio, ensuring every app gets its required regular scans for audit readiness.

What Changes When You Connect

01 You don't waste time manually exporting vulnerability reports. By using `list_issues` and `get_issue`, your agent compiles all the data you need into a clean summary, saving hours of spreadsheet work.

02 Start new audits on demand. Instead of navigating to the web console, just ask your agent to run a DAST scan using `start_dast_scan`. The whole process happens through conversation.

03 Get full visibility across all assets. You can use `list_apps` to see every single application ID in your inventory at a glance, ensuring no critical piece of software is forgotten during an audit.

04 Check the status without logging in. Need to know if last night's scan finished? Use `get_scan` and `list_scans` to get instant updates on running or completed jobs.

05 Manage internal systems easily. The `list_presence` tool shows you which local agents are available, letting you plan scans for apps that don't have a public URL.

Real-World Applications

Pre-Compliance Audit Check

A compliance officer needs to prove that all 40 internal applications were scanned this quarter. They ask their agent to run `list_apps` first, then use `list_scans` for each app ID to confirm coverage and gather proof of regular auditing.

Immediate Flaw Discovery

A developer asks the agent to check a newly deployed service. The agent uses `start_dast_scan`, waits for completion, and then runs `list_issues` to immediately report any high-severity flaws found.

Deep Dive into One Vulnerability

A security engineer finds a suspicious vulnerability ID. They ask the agent to run `get_issue` with that ID. The tool returns detailed context, including remediation steps and severity scores, allowing for immediate triage.

Inventory Cleanup

An ops team member suspects an old application is forgotten. They use `list_apps` to verify the existence of the app ID, then run `get_app` to check its details before deciding if it needs to be decommissioned.

Patterns to Avoid

Assuming AppScan knows everything

✗ AVOID

Telling your agent 'Tell me about the security of the Customer Portal.' The agent can't guess; it needs specific instructions and IDs.

✓ INSTEAD

First, use `list_apps` to get the exact app ID. Then, tell your agent: 'Use this ID with `get_app`, then run `list_issues` for that result.' This gives you targeted data.

Running scans without knowing targets

✗ AVOID

Just telling the tool to 'Scan everything.' The process fails because it needs a specific URL or application ID to start DAST.

✓ INSTEAD

Use `list_apps` to find the correct target. Then, initiate the scan by explicitly asking your agent: 'Start a new DAST scan for this app with this URL,' triggering `start_dast_scan`.

Overloading the chat session

✗ AVOID

Asking to list all apps, then check 20 scans, and finally start 5 new ones in one prompt. The agent gets bogged down.

✓ INSTEAD

Break it up. Use `list_apps` first. Then, dedicate a separate turn to checking the status of the most critical scan using `get_scan`. Keep your requests focused.

The Right Fit

Use this MCP if you manage security across numerous applications and need to automate repetitive auditing tasks—like listing vulnerabilities or starting scans—without leaving your chat window. It's built for deep, technical security work. Don't use it if you simply need a high-level dashboard summary of risk; the tool provides granular data that requires interpretation (e.g., using `get_issue` to understand severity). If you only need simple compliance reports based on date ranges and don't care about application IDs, a general reporting tool might suffice. But when your job revolves around checking specific vulnerabilities or managing complex scan cycles, this MCP is essential.

Security Audits Used to Be a Dashboard Nightmare

Right now, if you want to audit an application's security status, you open the AppScan dashboard. You manually select the app. Then you check the scan history. If things look good, great; if they don't, you have to export the vulnerability list into a spreadsheet. Then you copy that data into your compliance report. That cycle takes time and introduces manual error at every step.

With this MCP, you just talk to your agent. You ask it to audit the application inventory, and it runs `list_apps` and immediately gives you all the IDs. When you need flaw details, it uses `list_issues` and presents a clean list—no exporting needed. The whole process stays within conversation.

Get Vulnerability Data with HCL AppScan MCP

You ditch the constant clicking between tabs, the copy/paste routine of moving findings from one sheet to another, and the headache of cross-referencing different scan reports.

Now, when you ask your agent to get detailed vulnerability information using `get_issue`, the data comes back structured, actionable, and ready for immediate decision-making. It's a massive shift in workflow.

HCL AppScan MCP: 10 Tools for Security Auditing

These tools let you automate complex security tasks, from listing every application in your inventory to initiating a full dynamic analysis scan.

#	TOOL	DESCRIPTION
01	`get_account_check`	Verifies that your AppScan account connection is active and ready to use.
02	`get_account_info`	Retrieves specific information about the user who authenticated to the service.
03	`get_app`	Pulls detailed data for one specific application within your inventory.
04	`get_issue`	Gathers granular information about a single, identified vulnerability or security issue.
05	`get_scan`	Retrieves the status and details for one specific scan job.
06	`list_apps`	Shows a comprehensive list of every application registered in your AppScan inventory.
07	`list_issues`	Generates a full listing of all vulnerabilities found for a given application ID.
08	`list_presence`	Lists the local agents, or Presences, that are available to scan internal applications.
09	`list_scans`	Provides a complete record of all scans that have ever been run in your account history.
10	`start_dast_scan`	Begins a new Dynamic Analysis (DAST) scan for a specified web application URL.

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 in my AppScan inventory.

I've retrieved your application inventory. You have 15 applications, including 'Customer Portal', 'Payment API', and 'Mobile Backend'.

U Show me high severity issues for application 'Customer Portal'.

I've found 3 high severity issues for 'Customer Portal', including a 'SQL Injection' vulnerability and a 'Cross-Site Scripting' (XSS) issue.

U Start a new DAST scan for appId '12345' with URL 'https://portal.example.com'.

A new DAST scan (ID: scan_998877) has been successfully started for 'Customer Portal' targeting 'https://portal.example.com'.

Frequently Asked Questions

01 **How do I list all applications with HCL AppScan MCP?**

You simply ask your agent to list the apps using `list_apps`. This tool immediately shows you every application ID currently tracked in your security inventory.

02 **Can I start a scan without knowing the URL? (HCL AppScan MCP)**

No. The `start_dast_scan` tool requires a specific URL to run the dynamic analysis test. You must first find the target URL and pass it to the agent.

03 **What if I need details on one vulnerability? (HCL AppScan MCP)**

You use `get_issue` and provide the specific ID of the issue you care about. The tool returns detailed context, including severity and how to fix it.

04 **Does HCL AppScan MCP track old scans? (HCL AppScan MCP)**

Yes. You can use `list_scans` or `list_issues` to view historical data, helping you audit past performance and ensure compliance over time.

05 **What is the difference between listing apps and getting app details? (HCL AppScan MCP)**

Using `list_apps` gives a simple roster of all IDs. Using `get_app` retrieves deep, detailed information for one specific app ID you've already identified.

06 **How do I get my AppScan API Key ID and Secret?**

Log in to the AppScan on Cloud console, go to your **User Profile** (top right), and select **API Keys**. You can generate a new Key ID and Key Secret there.

07 **Does this server support the EU region?**

Yes, you can configure the `APPSCAN_REGION` environment variable to `eu` to connect to the European data center (`eu.cloud.appscan.com`).

08 **Can I start a scan for an internal application?**

Yes, provided you have an AppScan Presence (local agent) configured. You can use the `list_presence` tool to check their availability before starting a scan.

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 `"hcl-appscan": { "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

HCL AppScan 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 HCL AppScan. 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	HCL AppScan MCP
Server ID	`019d7551-001d-7171-b864-790c4c6e5e79`
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/hcl-appscan.