

M C P   S E R V E R

N O   C O D E

C L O U D   H O S T E D

# Apify MCP for AI Agents

Command web scraping and data extraction workflows through natural conversation.

Apify lets your AI agent run web scrapers, extract structured data, and manage browser tasks. It connects your Apify workspace to any MCP-compatible client so you can command complex data extraction workflows using natural language instead of writing custom scripts.

**A+** Quality Score 98.33/100

web-automation

data-extraction

proxy-services

headless-browser

datasets

api-integration



# 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](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Apify MCP

10 tools available

Cloud-hosted on Vinkius

Stop fighting with headless browser configurations and proxy rotations. This MCP gives your agent the ability to interact with your full Apify workspace directly. Instead of manually setting up actors or checking logs in a dashboard, you can just tell your agent to go grab a list of products from a specific site or pull the latest pricing data. It handles the heavy lifting of web scraping while you focus on what to do with the data. If you're already using Vinkius to manage your MCP catalog, adding this is the quickest way to turn your AI into a high-powered data gathering machine. You can start a scrape, check its progress, and grab the final JSON results without ever leaving your chat window. It's about moving from 'how do I scrape this' to 'give me the data from this site.'

---

## Core Capabilities

### 01 — Pull structured JSON

Get clean data from finished scraper runs directly into your agent's context.

### 02 — Kill runaway scrapes

Stop active jobs immediately to save on compute units and costs.

### 03 — Queue new URLs

Add new links to a scraper while it is already running to expand your crawl.

### 04 — Monitor account usage

Check your remaining compute units and subscription limits in real time.

### 05 — Fetch screenshots

Retrieve visual snapshots or configuration files from your actor runs.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/apify](https://vinkius.com/mcp/apify) — connect your AI agent in three steps.

- 01 Subscribe to the MCP and provide your Personal Apify API Token.
- 02 Connect the MCP to your preferred AI client like Claude or Cursor.
- 03 Command your agent to find actors, start runs, and pull datasets via chat.

The bottom line is you get a conversational interface for your entire Apify scraping infrastructure.

## Built For

This is for data engineers who need to automate extraction, market researchers gathering pricing, and developers who want to feed fresh web data into their own applications.

### Data Engineer

Triggers scheduled extractions and maps Apify objects during QA checks without writing extra glue code.

### Market Researcher

Commands the agent to scrape competitor prices and turn them into markdown tables for quick analysis.

### AI Developer

Feeds fresh, structured site data into their own agents using headless browser scrapers.

## What Changes When You Connect

- 01 Stop manual data entry by letting your agent pull JSON records directly into your workspace.
- 02 Control your costs by monitoring compute units and stopping runaway jobs instantly.

- 
- |    |                                                                                                     |
|----|-----------------------------------------------------------------------------------------------------|
| 03 | Expand your reach by pushing new URLs to active queues while a scraper is already running.          |
| 04 | Get visual proof of your scrapes by pulling screenshots from key-value stores into your chat.       |
| 05 | Save time on setup by using pre-built actors instead of writing custom scraping logic from scratch. |
- 

---

## Real-World Applications

### Competitive Price Tracking

A researcher asks the agent to scrape a list of electronics from a retail site. The agent runs the actor and returns a markdown table of prices.

### Data Enrichment

An engineer uses the agent to pull structured product profiles from a completed run to update a local database.

### Dynamic Crawling

A dev needs to scrape a site where links are generated on the fly. The agent finds the links and pushes them to the queue while the run is active.

### Resource Management

An ops lead asks the agent to check how many compute units are left for the month to plan the next big data pull.

---

## Patterns to Avoid

### Running long scrapes synchronously

#### ✗ AVOID

Trying to use a synchronous run command for a job that takes over five minutes.

#### ✓ INSTEAD

Use the asynchronous run tool to get a run ID, then poll the status until it finishes.

### Ignoring compute limits

#### ✗ AVOID

Firing off massive scrapes without checking your subscription balance first.

#### ✓ INSTEAD

Check your account limits before starting a large job to avoid unexpected overage charges.

---

## Hardcoding all URLs

### X AVOID

Providing a massive list of URLs upfront instead of letting the scraper discover them.

### ✓ INSTEAD

Start with a few seed URLs and use the queue tool to add new links as the agent discovers them.

---

## The Right Fit

Use this MCP if you need to scrape dynamic websites, bypass bot detection, or manage a fleet of complex scrapers via an AI agent. It's perfect for market research, lead generation, and data mining where the site structure is complex. Don't use this if you only need to grab a single static page from a simple URL; a basic fetch tool is faster and cheaper for that. Use this when you want to manage your entire Apify workspace and monitor your compute budget through a chat interface.

---

---

## Stop Manual Data Entry with the Apify MCP for Web Scraping

Most teams waste hours copy-pasting prices from websites into spreadsheets. You open a tab, scroll, highlight, copy, and paste. If the site layout changes, your process breaks. You're stuck doing the grunt work that a machine should handle.

With this MCP, you just tell your agent to go grab the data. It triggers the right actor, waits for the results, and hands you a clean JSON file or a markdown table. You go from manual clicking to instant data.

---

---

## Scale Your Research with the Apify MCP for Data Extraction

Scaling data collection usually means hiring more people or writing complex Python scripts to manage proxies and headers. It's a constant maintenance headache that drains your engineering resources.

This MCP lets you scale by just describing the goal. Your agent handles the execution logic across your Apify account. You get more data with less overhead.

---

# Apify MCP: 10 Tools for Web Scraping

Manage actors, monitor limits, and extract datasets with these 10 web scraping tools.

| #  | TOOL                             | DESCRIPTION                                                                                                    |
|----|----------------------------------|----------------------------------------------------------------------------------------------------------------|
| 01 | <code>abort_run</code>           | Stop an active Apify actor run. Use this to end runaway scrapes while preserving already collected data.       |
| 02 | <code>get_account_limits</code>  | Check your Apify account subscription limits. Monitor compute unit usage to avoid overage charges.             |
| 03 | <code>get_dataset_items</code>   | Export structured JSON data from an Apify dataset. Use limit and offset for paginated bulk downloads.          |
| 04 | <code>get_key_value_store</code> | Retrieve an item from an Apify actor key-value store. This is useful for getting screenshots or config files.  |
| 05 | <code>get_run</code>             | Check the status and metadata of a specific Apify actor run. Poll this to track long-running scrapes.          |
| 06 | <code>list_actors</code>         | List all accessible actors in your Apify account. See both owned actors and saved public templates.            |
| 07 | <code>list_webhooks</code>       | List all configured webhooks in your account. See target URLs and associated actor IDs for event-driven tasks. |
| 08 | <code>push_to_queue</code>       | Dynamically push new URLs to an active Apify request queue. This lets you add discovered pages mid-run.        |
| 09 | <code>run_actor_sync</code>      | Run an Apify actor and block until completion. This is best for short-lived tasks under five minutes.          |
| 10 | <code>run_actor</code>           | Start an Apify actor asynchronously with custom JSON input. Use this for large scrapes to get a run ID back.   |

---

## See It in Action

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

**U** List all the Apify actors available on my account.



I've scanned your Apify actor catalog. You possess 3 main actors: 'ecommerce-spider-v1' and two public templates 'google-search-scraper' and 'instagram-scraper'. Shall I print their base configurable inputs?

**U** Verify the status of run 'qKpwH9LgC3r0Xm' and show me its final dataset if finished.



I checked run `qKpwH9LgC3r0Xm`. Its current status is `SUCCEEDED`. The run consumed 0.045 Compute Units. I have successfully downloaded the associated dataset ( `ds_8aBxP_...` ) which yielded 302 product profiles. I'm injecting the top 5 parsed items below.

**U** How are our compute usage limits tracking this current month on Apify?



I pulled your overall compute records. Right now on the 'Scale' plan, your account used 82 out of 100 Compute Units (CU). You've consumed significant resources on proxy bandwidth (1.8TB/2TB) leading to roughly 80% usage threshold. I recommend holding further massive scraped runs until renewal happens on the 10th.

---

## Frequently Asked Questions

### 01 Can I use the Apify MCP to scrape social media sites?

Yes, you can use your agent to run specific actors designed for social platforms. The MCP allows your agent to trigger these runs and pull the resulting data back into your chat.

---

**02 Does the Apify MCP support bypassing bot detection?**

Yes, because it connects to your Apify workspace, it uses Apify's proxy networks and browser management to handle bot detection for you.

---

**03 How do I see how many compute units I have left with Apify MCP?**

You can simply ask your agent to check your account limits. It will pull your current subscription data and tell you how much of your monthly quota remains.

---

**04 Can my agent find new URLs while a scrape is running?**

Yes, the MCP supports pushing new URLs to an active queue. This allows your agent to manage dynamic crawls where new pages are discovered as the scraper moves.

---

**05 Is it possible to get screenshots of a website using the Apify MCP?**

Yes, you can ask your agent to retrieve items from your key-value store. This includes visual snapshots and configuration files from your completed runs.

---

**06 How can the AI agent run a scrape on a list of product URLs?**

First, find your specific scraping Actor ID via ``list_actors``. Then, prompt your agent to execute ``run_actor``, providing the target URLs formatted as a structured JSON input payload. It returns a 'Run ID'. You can poll this run via ``get_run``, and once it succeeds, the agent calls ``get_dataset_items`` to pull all acquired data straight to your window.

---

**07 Can the agent interact with run configurations mid-way during crawling?**

Yes. If an Apify crawler is currently executing and utilizes a Request Queue, you can instruct your agent to call ``push_to_queue``. Doing so dynamically ships new URLs to the active queue instance, extending the current web crawl without needing to stop or restart the Actor.

---

**08 Can my AI automatically detect scraping timeouts and debug the failure?**

Absolutely. Because your agent can track real execution flows with ``get_run``, it's aware if it transitions to TIMED-OUT or FAILED states. Subsequently, you can ask the agent to examine the KV Store log outputs ensuring the underlying issue (e.g. captcha block, blocking proxy) is identified immediately.

---

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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 <code>"apify": { "url": "..." }</code>       |
|  Windsurf | MCP Settings → <code>mcp_settings.json</code> → 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

# Apify 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](https://vinkius.com) · [support@vinkius.com](mailto: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 Apify. 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 | Apify MCP                                         |
| Server ID  | 019d754f-450f-733b-ad5f-50479ed751b0              |
| Platform   | Vinkius Cloud for AI Agents                       |
| Endpoint   | <code>https://edge.vinkius.com/{token}/mcp</code> |

**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/apify](https://vinkius.com/mcp/apify).