MCP SERVER | NO CODE | CLOUD HOSTED

WebScrapingAPI MCP

Structured Web Data Extraction Via Chat.

WebScrapingAPI delivers industrial-grade web data extraction directly through your AI client. Scrape raw HTML from any URL using datacenter proxies. Capture complex JavaScript rendering by running dynamic pages through a headless browser. Pull structured results, like product prices and search engine snippets (Google, Bing, Yandex), all via natural conversation.

A+ Quality Score 100/100

html-parsing | proxy-rotation | javascript-rendering | data-extraction | serp-data | headless-browser

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/webscrapingapi

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.

WebScrapingAPI MCP

10 tools available
Cloud-hosted on Vinkius

Stop copy-pasting data from websites into spreadsheets. This MCP lets you treat the live internet as a database. You talk to your agent—whether it's in Cursor or Claude—and tell it what you need: 'Give me the price and review count for this product on Walmart.' It handles the messy work of navigating dynamic sites, bypassing anti-bot measures with residential proxies, and structuring the data automatically.

It doesn't matter if a site loads its content using JavaScript or if you just need basic raw HTML. You simply ask your agent to scrape it, and it gets the full rendered state. Need competitive pricing? Ask for product details from major e-commerce sites like Amazon into clean JSON format. Want to know what people are searching for? It gathers structured search results across Google, Bing, and Yandex.

Connecting this MCP through Vinkius means you get access to all these capabilities—from basic scraping to advanced data parsing—all in one place. Your AI client becomes a web architect, gathering massive datasets or verifying competitor pricing without you ever writing a single line of scraper code.

Core Capabilities

01 — Extract Product Data

Pull structured details like price, title, and reviews from major e-commerce sites into clean JSON.

02 — Handle Dynamic Pages

Capture the full content of modern websites that rely on JavaScript to load their data using a headless browser.

03 — Scrape Raw Content

Retrieve basic HTML structure from any target URL using high-capacity datacenter proxies.

04 — Get Search Results

Fetch structured results, including organic listings and ads, from Google, Bing, or Yandex search engines.

05 — Customize Scrape Parameters

Run scrapes with advanced targeting options like specific geographical locations, session management, or custom headers.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to the WebScrapingAPI MCP and input your unique API key into Vinkius.

02 Activate the connector in your AI client (e.g., Cursor).

03 Ask your agent a question like, 'Find all product details from this Amazon link,' and it runs the scrape.

The bottom line is you talk to your agent naturally, telling it what data you need, and the MCP handles the complex process of accessing, rendering, and structuring that information for you.

Built For

This connector is built for people who spend too much time copying data across tabs. Think e-commerce analysts watching competitor pricing or SEO specialists tracking search snippet changes. If your job involves getting structured data from the live web, this is for you.

E-commerce Analyst

Monitors competitor pricing and product availability across multiple online marketplaces by running targeted scrapes.

Data Scientist

Collects massive, diverse datasets for machine learning training or market research using simple chat prompts instead of complex scripts.

SEO Specialist

Tracks search engine result page (SERP) rankings and analyzes structured snippets from different regions across Google, Bing, and Yandex.

What Changes When You Connect

01 Stop worrying about JavaScript. If a website loads its content dynamically, the `scrape_js_rendered` tool uses a headless browser to capture every element, ensuring you get complete data.

02 Never hit a block again. Use residential proxies via the `scrape_via_residential_proxy` tool for high anonymity and reliable access that bypasses aggressive bot detection.

03 Go beyond simple HTML dumps. The `scrape_ecommerce_product` tool automatically extracts key product details like price, title, and ratings into structured JSON format.

04 Get comprehensive search data from multiple engines. You can compare results by running `search_google_serp`, `search_bing_serp`, or `search_yandex_serp` against the same query.

05 Save time on complex requests with advanced options. The `custom_api_scrape` tool lets you specify geo-targets, sessions, and headers to fine-tune every scrape exactly how you need it.

Real-World Applications

Monitoring Competitor Pricing

An e-commerce manager wants to know if three rivals changed their price on Amazon. They ask their agent to run `scrape_ecommerce_product` against several competitor URLs, getting a clean JSON object for each one instantly.

Analyzing Search Trends

An SEO specialist needs to see how 'best running shoes' ranks across different search engines. They ask their agent to run `search_google_serp` and then follow up with `search_bing_serp` to compare the structured results.

Gathering Market Research Data

A data scientist needs a dataset of news articles. They tell their agent to run `scrape_and_auto_extract` on 50 different news links, getting clean, usable content without manual parsing.

Testing Dynamic Web Applications

A developer needs to verify if a complex dashboard page works correctly. They use the agent and request `scrape_js_rendered` on the test URL, confirming all elements are captured.

Patterns to Avoid

Using simple scraping for modern sites

✗ AVOID

Trying to scrape a complex Single Page Application (SPA) using basic methods and only getting incomplete or empty results.

✓ INSTEAD

Instead, use the `scrape_js_rendered` tool. This simulates a full browser session, guaranteeing that all JavaScript-loaded content gets captured before extraction.

Ignoring proxy restrictions

✗ AVOID

Running multiple scrapes from your usual IP address and getting blocked by the target website's security measures.

✓ INSTEAD

Use `scrape_via_residential_proxy`. This tool cycles through real residential addresses, making your scraping look like traffic from diverse, legitimate users.

Manually selecting data points

✗ AVOID

Having to write custom selectors or CSS paths for every product page on Amazon when you just need the price and title.

✓ INSTEAD

Use `scrape_ecommerce_product`. This tool handles the specific structure of major e-commerce sites, automatically returning only the core details in a clean JSON format.

The Right Fit

You should use this MCP if your goal is data extraction from live websites that aren't simple static pages. If you need to scrape JavaScript content, you must use `scrape_js_rendered` or related tools. If the target site has robust anti-bot measures, you need to utilize residential proxies via `scrape_via_residential_proxy`. Don't use this if you only need data from a known API endpoint; those services are better for that. Also, don't use it if your goal is just reading text—use the structured tools like `scrape_ecommerce_product` or `search_google_serp` to get usable JSON output instead of raw HTML.

Getting Data from Websites Is a Messy Chore.

Right now, collecting market data feels like a digital scavenger hunt. You open 30 tabs: one for Amazon, one for Walmart, and five different news sites. Then you copy the price from Tab A into Excel, paste the review count from Tab B next to it, and manually type up the structured search results you found in Google’s SERP. It's slow, tedious, and highly prone to human error.

With this MCP, your agent handles all that clicking and copy-pasting for you. You ask for 'product price, title, and review count from these five links.' The system runs the necessary scrapes—handling dynamic content, bypassing blocks, and extracting structured data—and delivers one perfectly formatted JSON payload. It’s instant.

Structured Web Data Extraction with WebScrapingAPI

The manual steps that vanish are the proxy management, the JavaScript debugging, and the inconsistent data formatting. You don't spend time figuring out if a site is static or dynamic; you just ask for the content.

What changes is your focus. Instead of managing web scraping infrastructure, you use AI to analyze what you've gathered. It makes all the difference.

WebScrapingAPI: 10 Tools for Web Extraction

These tools let you perform every type of data extraction imaginable—from raw HTML dumps to highly structured e-commerce product records, all controlled by your AI agent.

#	TOOL	DESCRIPTION
01	`custom_api_scrape`	Execute a scrape with advanced options like geo-targeting, specific sessions, or custom headers.
02	`scrape_and_auto_extract`	Scrape news or product pages and automatically extract structured data from the content.
03	`scrape_ecommerce_product`	Extracts price, title, and reviews as clean JSON format from Amazon, Walmart, or similar stores.
04	`scrape_static_html`	Retrieves the raw HTML structure from any target URL using datacenter proxies.
05	`scrape_js_rendered`	Scrapes dynamic pages by simulating a full headless browser render, capturing all JavaScript-generated elements.
06	`scrape_as_mobile`	Runs the scrape pretending it's coming from a mobile device to mimic real user access patterns.
07	`scrape_via_residential_proxy`	Uses residential proxies for high anonymity, helping bypass aggressive bot detection systems.
08	`search_bing_serp`	Retrieves structured search results data specifically from Bing.
09	`search_google_serp`	Gets structured search engine results from Google based on a provided query string.
10	`search_yandex_serp`	Retrieves structured search engine results data specifically from Yandex.

See It in Action

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

U Scrape the rendered HTML of 'https://example.com/dynamic-dashboard'.

I've scraped the dynamic page using a headless browser. Here is the rendered content, including all JavaScript-generated elements and data points from the dashboard.

U Search Google for 'best wireless noise cancelling headphones' and return structured results.

I found the top Google SERP results for your query. 1. 'Sony WH-1000XM5 review' (https://...), 2. 'Bose QuietComfort Ultra' (https://...), 3. 'Apple AirPods Max' (https://...). Would you like to see the snippets or ads associated with these?

U Get the price and rating for the product at 'https://amazon.com/dp/B09XXX'.

Successfully extracted product data from Amazon: Name: 'AcousticPro Wireless', Price: '$299.00', Rating: '4.8/5 stars (1,250 reviews)'. Would you like the full JSON extract?

Frequently Asked Questions

01 **Can WebScrapingAPI handle JavaScript-rendered websites?**

Yes, absolutely. Use the `scrape_js_rendered` tool. This function simulates a full browser to ensure that all content loaded by JavaScript is captured in your scrape.

02 **How do I get structured search results using WebScrapingAPI?**

You use the dedicated SERP tools like `search_google_serp` or `search_bing_serp`. These functions pull clean, structured data on ads and organic listings, not just a text summary.

03 **Is WebScrapingAPI good for scraping e-commerce sites?**

It's excellent. The `scrape_ecommerce_product` tool is designed specifically to pull critical product details, like price and review scores, into easily usable JSON format.

04 **What if my scrapes get blocked by the site?**

Use `scrape_via_residential_proxy`. This feature cycles through real user IP addresses, giving you high anonymity that helps bypass aggressive bot detection systems.

05 **Does WebScrapingAPI support different countries/regions?**

Yes. You can use `custom_api_scrape` to execute scrapes with specific geo-targeting options, allowing you to monitor data from anywhere in the world.

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 `"webscrapingapi": { "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

WebScrapingAPI 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 WebScrapingAPI. 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	WebScrapingAPI MCP
Server ID	`019d7621-888d-7307-9384-43e52f233903`
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/webscrapingapi.