MCP SERVER | NO CODE | CLOUD HOSTED

Browse AI MCP for AI Agents

Automating Web Data Extraction and Site Monitoring

Browse AI connects web scraping, data extraction, and website monitoring into a single conversational tool for your AI agents. Instead of building complex pipelines, you can ask your agent to track competitor prices, gather product details from multiple pages, or monitor changes on key industry sites—all through natural language conversation.

A+ Quality Score 100/100

data-extraction | web-monitoring | no-code | robot-automation | data-retrieval | scraping-tasks

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

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.

Browse AI MCP

10 tools available
Cloud-hosted on Vinkius

Need reliable data pulled off the web without writing code? This MCP lets you talk to your agent and get structured information straight into your workflow. You can tell it to check a website for specific price points, run automated reports across multiple sources, or list every robot scraper you own. It handles the complicated part of managing scraping jobs—from triggering runs on URLs to pulling back the final data.

If you're used to building complex scripts just to monitor a few prices, this changes that. You simply connect your agent via Vinkius and ask for what you need. Your agent then uses these tools to manage monitors, run tasks against specific websites, and retrieve structured data into a format ready for analysis. It's all about automating the tedious work of web data capture.

Core Capabilities

01 — Manage Web Scrapers

Get a list of your existing scraping robots and view detailed information about each one.

02 — Execute Data Capture Tasks

Initiate data extraction runs on specific web pages, then monitor the task status until the structured results are ready for you to pull.

03 — Monitor Websites Automatically

Set up scheduled alerts that track changes on any given website, so you don't have to check it manually every day.

04 — Review Bulk Data Jobs

View and manage runs where data was extracted from multiple sources simultaneously, giving you a high-level overview of all the captured information.

05 — Check System Health

Verify the operational status of the web scraping infrastructure and check current task queues.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and supply your unique Browse AI Secret API Key.

02 Connect your agent via any MCP-compatible client, granting it access to the web data tools.

03 Ask your agent to perform an action, like 'Run a monitor on the competitor's pricing page,' and retrieve the results.

The bottom line is: You use natural language conversation to control powerful, underlying web automation processes.

Built For

This MCP is for the data analyst who hates manual dashboard exports and spends hours copy-pasting competitor prices. It's for the growth marketer needing real-time alerts on product changes or pricing shifts. And it’s for developers who want to bake robust web extraction logic into their existing code base without writing boilerplate API handling.

Market Research Analyst

Uses the MCP to list monitors and set up schedules, ensuring they get automatic alerts when a competitor changes its product lineup or pricing structure.

E-commerce Growth Marketer

Triggers data extraction tasks on specific URLs to pull in current inventory levels or new listing details, feeding that fresh data directly into their campaign dashboard.

Software Developer

Integrates web extraction logic using the MCP's tools, allowing them to run robots and handle bulk runs from within a coding environment via natural language prompts.

What Changes When You Connect

01 Stop exporting dashboards manually. Use 'get_task' to pull structured data directly into your conversation, eliminating manual copy-pasting.

02 Monitor competitors effortlessly. Set up a monitor schedule using 'create_monitor' so your agent alerts you immediately when prices or listings change.

03 Manage complex operations easily. Run multiple sources at once and view the results by calling 'list_bulk_runs', keeping all your data in one place.

04 Control every aspect of scraping. From listing available scrapers with 'list_robots' to initiating a run via 'run_robot,' you control the whole process conversationally.

05 Check infrastructure health instantly. Use 'get_system_status' when you need to know if the web data tools are running smoothly or if there's a queue backup.

Real-World Applications

Tracking Competitor Pricing Changes

A growth marketer asks their agent, 'Set up monitoring for competitor prices.' The agent uses the `create_monitor` tool to track specific URLs, alerting the user immediately if a price changes by more than 10%.

Gathering Lead Information from Job Boards

A data analyst tells their agent to 'Run the lead extractor robot on today's top job board.' The agent uses `run_robot` and then retrieves the captured structured data via `get_task`, giving a clean list of contact details.

Auditing Multiple Website Data Sources

A researcher needs to compare product specs across five different vendor sites. They ask their agent to manage and inspect the results using `list_bulk_runs` after running several tasks, comparing all data points at once.

Verifying Scraping Capability

A developer needs to know if their scraping setup is working. They ask the agent for a system check, and it uses `get_system_status` to confirm that the whole web data infrastructure is online.

Patterns to Avoid

Ignoring Monitor Schedules

✗ AVOID

Assuming your monitoring tasks run automatically without checking if they are properly configured or listed.

✓ INSTEAD

Always use the `list_monitors` tool to verify which robot is tracking what, and ensure you set up new schedules using `create_monitor`.

Confusing Tasks with Robots

✗ AVOID

Trying to get data from a task without knowing which specific robot created it or running the necessary retrieval command.

✓ INSTEAD

First, use `list_robots` to confirm your scraper is ready. Then, call `run_robot` to create a task ID, and finally use `get_task` with that ID to pull the results.

Overlooking Bulk Runs

✗ AVOID

Running multiple small extraction tasks individually when you actually needed one comprehensive report across many pages.

✓ INSTEAD

Use `list_bulk_runs` and `get_bulk_run` to manage large data pulls. This function is built for extracting data from multiple sources simultaneously.

The Right Fit

Use this MCP if your job requires pulling structured, actionable information from websites that aren't already in a clean API format. You need continuous monitoring (using `create_monitor`) or massive batch processing of web pages (using bulk runs). Don't use it if you only need to pull data from a single, well-documented source using an existing API key; those services are usually faster and more direct. If your goal is purely internal database management without external web sources, look at dedicated backend connectors instead.

Browse AI MCP for Web Data Extraction: Automating Market Research Scraping

Right now, gathering competitive intelligence means jumping between tabs: visiting a product page to check the price, switching to another site to check stock levels, and then manually compiling a spreadsheet of discrepancies. This process is slow, prone to copy-paste errors, and impossible to scale.

With this MCP, you tell your agent exactly what data points are important—say, 'Product Name,' 'Price,' and 'Availability.' Your agent handles the web scraping logic using the tools, running tasks against specific URLs. You get a clean, structured list of comparative data pulled directly into your chat window.

Browse AI MCP for Site Monitoring: Tracking Website Changes Automatically

The old way required setting up multiple calendar reminders to check if a competitor changed their sales banner or added new service offerings. You'd have to manually inspect the site daily, losing valuable time in the process.

Now, you simply ask your agent to 'Monitor this URL for changes.' The MCP uses `create_monitor` under the hood. It autonomously tracks specified elements and alerts you only when a change occurs, saving you hours of manual website inspection.

Browse AI: 10 Tools for Web Data Extraction and Monitoring

Use these tools to manage robot lifecycles, schedule website monitors, execute data capture tasks, and retrieve structured web data reports.

#	TOOL	DESCRIPTION
01	`create_monitor`	Sets up a scheduled task that automatically tracks changes on a designated website for a specific robot.
02	`get_bulk_run`	Retrieves detailed information about one specific bulk data extraction job.
03	`get_robot`	Fetches the full metadata and details for a single, named web scraping robot.
04	`get_system_status`	Checks the overall operational status of the browsing AI infrastructure and its current task queue.
05	`get_task`	Retrieves the final data and status report for a single, ongoing web scraping task.
06	`list_bulk_runs`	Provides an overview of all previously executed bulk data extraction jobs across various sources.
07	`list_monitors`	Shows a list of all existing, scheduled website monitors tied to a specific robot.
08	`list_robots`	Retrieves the names and metadata for every approved web scraping robot you have set up.
09	`list_tasks`	Lists all tasks associated with a specific robot, helping you track multiple running jobs.
10	`run_robot`	Starts a data extraction process by running a specified web scraping robot against a target URL, creating a new task.

See It in Action

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

U List all my approved web scraping robots.

✅ **Your Robots:**

- Amazon Price Tracker (ID: rob_1)
- LinkedIn Lead Extractor (ID: rob_2)
- Real Estate Monitor (ID: rob_3)

These scrapers are active and ready to pull data. Which one should we run first?

U Run robot 'rob_2' on the new lead page and get me the data.

🚀 **Task Initiated:**
Robot `rob_2` is now processing the URL. We created task ID `task_998877`.

Status: Processing...
ETA: 5 minutes.

You can check progress or pull results using this ID.

U What data did task task_99283 pull?

✅ **Task Status: SUCCESSFUL**

Task `task_99283` is complete. Here's the structured summary:

FIELD	VALUE
Product Name	Modern Lamp
Price	$45.00
Stock	In Stock

Would you like me to generate a JSON file with all this data?

Frequently Asked Questions

01 **How can I automatically track changes on competitor websites using Browse AI?**

You can set up scheduled monitoring for specific pages. The system tracks defined elements (like pricing or listings) and alerts you instantly if the content deviates from what was previously captured. This saves you constant manual checks.

02 **Do I need to write code to extract data with Browse AI?**

No, you don't. You just talk to your agent and tell it which data points you want from a given URL or set of websites. The MCP handles the complex scraping logic in the background.

03 **What is the difference between running a task and bulk runs with Browse AI?**

A single 'task' extracts data from one specific URL at a time. Bulk runs, however, are designed to pull data across many different sources or pages simultaneously, giving you a comprehensive report.

04 **Can I use Browse AI MCP to gather leads from job boards?**

Yes. You can instruct your agent to run specific scraping robots on large websites like job boards. The MCP extracts the required data, such as names and contact info, into a usable list.

05 **Does Browse AI MCP work with my existing code?**

Absolutely. You can integrate web extraction logic directly into your coding environment by having your agent manage the tasks and retrieve results through conversational commands.

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 `"browse-ai": { "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

Browse AI 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 Browse AI. 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	Browse AI MCP
Server ID	`019d7564-2c3a-70c1-ac54-cc9e04cd833f`
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/browse-ai.