MCP SERVER | NO CODE | CLOUD HOSTED

Browserhub MCP for AI Agents

Automating large-scale web data extraction and proxy management

Browserhub provides AI agents with immediate access to enterprise web scraping capabilities. Your agent can run structured scrapers or perform one-off URL extractions using real browsers. It manages job monitoring, tracks task history, and controls proxy rotation so you get reliable, clean data directly into your workflow.

A+ Quality Score 100/100

proxy-management | anti-bot | javascript-rendering | data-extraction | web-scraping | automation

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

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.

Browserhub MCP

10 tools available
Cloud-hosted on Vinkius

Connecting Browserhub lets your AI client manage complex web scraping tasks through simple conversation. Instead of writing boilerplate code for every website structure, your agent simply tells the MCP what to extract from a given URL or list of URLs. It handles everything needed—from managing proxy locations and checking account credits to launching and monitoring multi-step scraping jobs.

If you're working with large datasets, this MCP ensures your data extraction remains reliable, even if sites change their layout. You can instruct your agent to check which scrapers are already set up or trigger a brand new job right out of the box. Because Vinkius hosts all these connections in one place, you get access to industrial-strength web automation without needing multiple specialized services. The result is clean, structured data waiting for you to analyze.

Core Capabilities

01 — Get Account Credit Balance

Checks your remaining account credit balance to ensure you have enough funds for scraping jobs.

02 — List All Scrapers and Blueprints

Retrieves comprehensive lists of all scrapers and blueprints configured in the system.

03 — Run a Defined Scraper Job

Starts a scraping job using an existing, named scraper blueprint for consistent data retrieval.

04 — Directly Scrape a URL

Performs immediate extraction on a single web page without needing to select or configure a specific scraper first.

05 — Manage Proxy Locations

Lists all available proxy locations, which helps prevent IP bans during large-scale scraping operations.

One Click on Vinkius — From Prompt to Execution

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

01 First, connect your AI client to this MCP using your Browserhub API Key.

02 Next, instruct your agent on the task—for example, 'Scrape these 5 URLs' or 'Run my E-commerce scraper for today's prices.'

03 The system executes the job and returns structured data, which your agent can then process directly into your workspace.

The bottom line is you get automated web data collection without writing custom API calls or managing proxy lists manually.

Built For

This MCP targets data-heavy roles—the market researcher who needs to track competitor pricing daily, the analyst building large datasets for machine learning, and developers integrating web crawling into production applications. If your job requires gathering external website data, this is built for you.

Market Researcher

Monitors competitor websites to track pricing changes, product availability, or news sentiment across multiple domains.

Data Scientist

Builds large-scale datasets for training models by automatically gathering structured information from diverse web sources.

Software Developer

Integrates reliable, proxy-managed web extraction into application backends using natural language prompts.

What Changes When You Connect

01 Monitor ongoing jobs with the `get_scraping_job` tool. You always know if your scraping task is running, stalled, or ready for data retrieval.

02 Don't waste time setting up scrapers from scratch. Use `list_scrapers` and `list_blueprints` to see everything you already have configured right away.

03 Need a quick look at one page? The `direct_scrape` tool lets your agent pull data immediately without needing pre-defined automation rules.

04 Stay safe from IP bans. Use the proxy management tools, like `list_proxy_locations`, to ensure every scraping session uses fresh, available IPs.

05 Always know your budget. Run `get_account_balance` whenever you start a large project so your agent doesn't hit an unexpected credit limit mid-run.

Real-World Applications

Tracking Competitor Pricing Changes

A market researcher needs to know if five key competitors changed their pricing on specific product pages. They tell their agent, 'Use the E-commerce scraper for the latest prices.' The agent runs the job and returns a clean spreadsheet of current vs. previous data.

Building Training Datasets

A data scientist needs thousands of structured articles on a specific topic. They ask their agent to list all available scrapers, pick one for 'News Articles,' and run it across 100 different domains using proxy rotation.

Real-time Content Monitoring

A developer wants to check if a key piece of content was updated on a partner site. Instead of building a whole scraper, they use the agent and ask it to `direct_scrape` that single URL, getting immediate validation.

Auditing Scraper Health

A developer suspects one of their existing scrapers is broken or outdated. They tell the agent to `get_scraper` details for a specific blueprint and check its metadata instantly before attempting a job run.

Patterns to Avoid

Trying to scrape everything with one giant prompt

✗ AVOID

Asking your agent, 'Scrape data from all websites, monitor jobs, and also check my credits.' The AI struggles because the requests are too diverse and lack specific structure.

✓ INSTEAD

Break it down. First, run `list_scrapers` to confirm what you need. Then, use a dedicated tool like `run_scraper` with a defined blueprint. Only after confirmation do you ask for the results using `get_scraping_job`.

Ignoring proxy limitations

✗ AVOID

Running multiple scraping jobs without checking IP availability, leading to job failures and potential account blocks.

✓ INSTEAD

Always check available infrastructure first. Use `list_proxy_locations` before starting any large batch of work to ensure you have a clean pool of IPs.

Not verifying the scraper setup

✗ AVOID

Assuming an old scraper blueprint still works because it was created months ago, leading to corrupted or incomplete data.

✓ INSTEAD

Before running any job, always use `get_blueprint` on the specific scraper ID. This confirms its current metadata and structure.

The Right Fit

Use this MCP if your primary bottleneck is gathering structured web content at scale. You need a system that manages proxy rotation and handles complex scraping jobs without you writing boilerplate code for each website's HTML structure. If you only need to send simple messages or manage contacts, don't use this; it's purely for data extraction. Don't use it if your goal is analyzing existing files in a database—you'd need an internal data warehouse connection instead. However, if the source of your truth is always 'the internet,' Browserhub gives you the necessary tools to reliably get that information.

Browserhub MCP for AI Agents: Solving Web Data Extraction Pain

Today, gathering data from websites means logging into multiple sites, manually copying tables and lists, and praying the website owner doesn't change a single element. You spend hours clicking through tabs and copy-pasting messy text that requires heavy clean-up later.

With this MCP, you simply tell your agent to extract what you need. It handles the entire process—running scrapers, managing proxies, and retrieving structured data directly into your workflow. You get clean, usable results without touching a single browser tab.

Browserhub MCP for AI Agents: Mastering Web Data Management

The pain points disappear when you can list all available scrapers and blueprints with one command. You don't have to remember which scraper ID was used last week or where the proxy IPs are located.

This gives your agent persistent, reliable access to structured data sources that were previously locked behind manual processes. Your workflow moves from tedious copying to instant, actionable insights.

Browserhub: 10 Tools for Structured Web Data Extraction

These tools give your agent granular control over the entire web scraping lifecycle, from listing scrapers to executing jobs and checking account status.

#	TOOL	DESCRIPTION
01	`direct_scrape`	Runs a quick scrape on a single URL without needing to create or select a specific scraper blueprint first.
02	`get_account_balance`	Checks how many credits are remaining in your Browserhub account.
03	`get_blueprint`	Retrieves specific details and metadata for a single, defined scraping blueprint.
04	`get_scraping_job`	Gets the status and final results of a previously run or currently running job.
05	`get_scraper`	Retrieves details about one specific, configured scraper setup.
06	`list_blueprints`	Shows a full list of every scraping blueprint you have set up in your account.
07	`list_scraping_jobs`	Lists the history and metadata for all past web scraping jobs.
08	`list_proxy_locations`	Shows every available proxy location you can use to prevent IP blocking during scraping.
09	`list_scrapers`	Lists all scrapers and their configurations that you have defined in your account.
10	`run_scraper`	Starts a new scraping job using one of your existing, named scraper blueprints.

See It in Action

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

U Check my account balance and list all my configured scrapers.

Account Status:
Credits Remaining: 1,500
Estimated Runs Left: ~200

Active Scrapers Found:
1. **E-commerce Prices (ID: scr_1):** Tracks competitor pricing from the top 3 retail sites.
2. **News Articles (ID: scr_2):** Captures headlines and authors from major news outlets.
3. **Lead Generator (ID: scr_3):** Scrapes contact details from professional networking profiles.

U Scrape the URL for the new quarterly report I just found.

✅ Scraping job `job_99283` started successfully for the specified URL.

I've initiated a real browser scrape. This takes time, but I'll monitor it and notify you immediately when the structured data—including tables and contact info—is ready to pull into your workspace.

U What are my scraping jobs history for last month?

Job History Summary:
- **job_98123 (Date: 6/10):** Status: Complete. Result: 4,500 records extracted.
- **job_97112 (Date: 5/28):** Status: Failed. Error: Timeout on specific domain.
- **job_96001 (Date: 5/1):** Status: Complete. Result: 1,200 records extracted.

Frequently Asked Questions

01 **How does the Browserhub MCP help me scrape multiple websites?**

It manages the complexity of web scraping by handling proxy rotation and IP management. You don't worry about being detected or blocked; the agent handles switching IPs automatically to keep your jobs running smoothly.

02 **Can I use Browserhub MCP if I don’t have a specific scraper set up?**

Yes, you can. The direct scraping capability lets your agent perform one-off extractions on any URL immediately, even if you haven't created a complex blueprint for it.

03 **Is this MCP better than writing custom Python web scrapers?**

For most people, yes. This MCP lets your agent manage the scraping process via natural language prompts, saving you hundreds of lines of code and maintenance time when websites change their structure.

04 **What if I need to check my current scrape job status?**

You simply ask your agent to check the job by referencing its ID. It will give you a real-time update on whether it's running, waiting for proxies, or fully finished.

05 **Can I list all my available scrapers using Browserhub MCP?**

Absolutely. The agent can list every scraper and blueprint you have configured in your account, giving you a quick overview of all your data collection assets.

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 `"browserhub": { "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

Browserhub 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 Browserhub. 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	Browserhub MCP
Server ID	`019d7564-a053-7253-8270-235f14bcf117`
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/browserhub.