MCP SERVER | NO CODE | CLOUD HOSTED

Docker Hub MCP for AI Agents

Manage container image tags and repositories efficiently

Docker Hub MCP lets your AI client manage container images directly from natural conversation. Check repository details, discover image tags with versions and sizes, or search thousands of public community images without logging into a web dashboard.

A+ Quality Score 100/100

containerization | registry | image-management | automation | repository | version-control

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

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.

Docker Hub MCP

10 tools available
Cloud-hosted on Vinkius

Managing containers usually means clicking through dozens of tabs to check pull counts or verify if the right version tag exists. This MCP lets you bypass the UI entirely. Your AI agent connects straight to your Docker Hub account, letting you perform registry tasks using simple conversation. Need to know what versions are available for a specific image? Just ask. Want to see how popular a repository is across organizations? It's ready. You can use it to list and manage repositories, browse all associated tags with version info, or even search the massive public catalog for community images. With Vinkius managing this connection, your AI agent acts like a dedicated DevOps assistant, handling everything from listing organization members to updating repository descriptions—all without you ever touching the Docker Hub website.

Core Capabilities

01 — List all repositories

Retrieves a list of image repositories belonging to your user or organization.

02 — Discover available tags for an image

Shows every version tag associated with a specific repository, including size and push date.

03 — Search public images by name

Finds matching community repositories across the entire Docker Hub catalog based on keywords or descriptions.

04 — Create new image repositories

Sets up a brand-new repository within your account, defining its visibility and description.

05 — Update existing repo details

Changes metadata on an established repository, such as updating the description or setting privacy status.

One Click on Vinkius — From Prompt to Execution

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

01 You subscribe to this MCP and provide your Docker Hub Access Token.

02 Your AI client authenticates with Vinkius and gains access to all available container registry tools.

03 You give a natural language command, like 'list the tags for my web service repo,' and your agent executes the action.

The bottom line is that you talk to your agent about your containers, and it handles the API calls necessary to get you the data or make the change.

Built For

This MCP is built for developers and operations engineers who spend too much time navigating UIs just to check image versions or repository status. It's ideal for anyone tired of context-switching between their terminal, a ticketing system, and the Docker Hub website.

DevOps Engineer

Checking pull counts on multiple repositories, auditing access, or listing organization membership to ensure compliance.

Software Developer

Quickly verifying if a specific image tag (like v1.2.3-alpine) exists before committing code, or searching for recommended community base images.

Platform Architect

Managing the lifecycle of core services by creating new repositories and ensuring correct visibility settings across organizations.

What Changes When You Connect

01 Checking available versions: Use the `list_tags` tool to instantly view every tag for a repo, eliminating manual browsing of version history.

02 Finding base images: The `search_repositories` tool lets you find community Docker images across the whole catalog using only keywords.

03 Auditing status: Run `list_repositories` and monitor pull counts or last updated dates without visiting the dashboard page.

04 Maintaining integrity: Use `get_repository` to instantly confirm metadata, like ownership or description, for any service repo.

05 Updating details: Never manually edit a repo again. The `update_repository` tool lets you change descriptions or privacy settings via command.

Real-World Applications

Verifying the correct image version

A developer needs to confirm if `nginx:1.25-alpine` is available for deployment. Instead of searching through dozens of tags, they ask their agent, and it uses `list_tags` to provide a clean list showing the size and push date.

Checking organizational access rights

An engineer needs to know which teams belong to their organization. The agent uses `list_organizations` and can then list all associated repositories, providing a quick audit trail for compliance checks.

Starting a new microservice container

A platform team needs a dedicated repo for a new service. They use the agent's command capability to execute `create_repository`, instantly setting up the required namespace and privacy settings.

Patterns to Avoid

Manually checking repo stats

✗ AVOID

Opening the Docker Hub website and clicking through multiple repository pages to check individual pull counts or last updated dates.

✓ INSTEAD

Use `list_repositories` to get a single, comprehensive overview of all your repositories' star count, pull count, and update date in one query.

Forgetting image versions

✗ AVOID

Assuming the 'latest' tag has the version I need, only to find out later that a critical bug was fixed in an older, specific release.

✓ INSTEAD

Use `list_tags` to browse all available tags for a repository and verify the precise, stable version you should be pulling.

Missing community images

✗ AVOID

Only remembering to search official documentation when looking for niche base images (e.g., a specific language runtime or OS variant).

✓ INSTEAD

Use `search_repositories` to query the entire public catalog, finding relevant community images by name or description instantly.

The Right Fit

Use this MCP if your workflow involves frequent checks of container image metadata, repository status, or version history. It's perfect for automating tasks like listing tags (`list_tags`), checking pull stats (`list_repositories`), and managing basic repo structure (like `create_repository`). Don't use it if you need to push actual images; this MCP only manages the *metadata* about the images. If your goal is complex build pipeline orchestration or interacting with source code repositories, you might need a specialized Git integration instead.

Docker Hub MCP: Automating Repository Metadata Management

Today, managing container assets means logging into the Docker Hub website. You'll navigate to your organization, click through repo lists, and then open each repository page just to check its last updated date or manually verify if a specific version tag exists before starting a build script.

With this MCP, you simply ask your agent for an overview of all repositories using `list_repositories`. The result instantly shows the name, star count, pull count, and visibility status for every service, giving you a full audit in one clean response.

Docker Hub MCP: Discovering Community Container Images

Finding niche base images manually is a huge time sink. You typically have to remember the exact name or browse through categories until you find that perfect language runtime or utility image.

Now, just ask your agent to search for what you need using `search_repositories`. It queries the whole public catalog and returns relevant community images instantly, saving you from hours of browsing.

Docker Hub: 10 Tools for Repository & Image Management

These tools allow your agent to perform specific actions like creating repos, listing tags, or searching the public catalog from a single interface.

#	TOOL	DESCRIPTION
01	`create_repository`	Creates a brand-new container repository, allowing you to set its description and whether it should be public or private.
02	`delete_repository`	Permanently removes an entire Docker Hub repository. Be careful; this action cannot be undone.
03	`get_repository`	Retrieves specific details, including ownership and creation metadata, for a given repository name.
04	`get_tag`	Fetches detailed information about a single image tag, providing its size and push date.
05	`get_user`	Confirms your connected identity by retrieving basic user info like name, email, and account type.
06	`list_organizations`	Shows all the organizational groups you are a member of on Docker Hub.
07	`list_repositories`	Provides an overview of all your repositories, including their star count and last update date.
08	`list_tags`	Lists every available tag for a specific repository, useful for seeing version history.
09	`search_repositories`	Searches the public Docker Hub catalog to discover community images based on keywords or descriptions.
10	`update_repository`	Modifies metadata for an existing repository, such as changing its description or setting its private status.

See It in Action

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

U I need a quick list of all my current production repositories.

Repositories Overview

- `api-service` : 12.5K pulls (Public) - Last updated: 3 days ago
- `web-frontend` : 8.9K pulls (Public) - Last updated: 1 week ago
- `internal-worker` : Private - Last updated: 2 months ago

You can also check the full details on any of these.

U Show me all available versions for my base image repo.

The `base-image` repository has several tags. Here are some recent ones:

- **latest**: 150MB (Pushed: Today)
- **v2.1.3**: 149MB (Pushed: 5 days ago)
- **v2.1-alpine**: 80MB (Pushed: 5 days ago)

The latest stable tag is `v2.1.3`.

U Search for a lightweight Python base image.

I found several matching images:

- **Official Python**: The main repository has over 1.2B pulls.
- **Python Alpine:** A great lightweight option (approx. 50MB).
- **Slimmed Version:** Also available, offering a minimal footprint for production builds.

Frequently Asked Questions

01 **How do I check my repository's pull count using the Docker Hub MCP?**

You can use this MCP to list all your repositories and see their pull counts instantly. This saves you from having to visit each repo page individually just for basic metrics.

02 **Can I find community images in the Docker Hub using this MCP?**

Yes, absolutely. You can use the search function within the MCP to query the entire public catalog of Docker Hub, finding niche base images or related projects by name or description.

03 **What if I need to change my repository's privacy status?**

The MCP includes a tool that lets you update your repositories. You can use it to switch a repo from public to private, or vice versa, directly through conversation.

04 **Does the Docker Hub MCP help me manage multiple organizations?**

Yes. The MCP allows you to list all the organizations you belong to and view repositories across those different organizational accounts in one place.

05 **What is the difference between listing tags and getting repo details with this MCP?**

Listing tags gives you a comprehensive list of every version tag for an image. Getting repository details provides high-level metadata, like who owns it or when it was last updated.

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 `"docker-hub": { "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

Docker Hub 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 Docker Hub. 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	Docker Hub MCP
Server ID	`019d842f-2716-70d5-915f-320712f6d4b2`
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/docker-hub.