MCP SERVER | NO CODE | CLOUD HOSTED

OpenSea MCP

Track NFT market value and ownership records.

OpenSea MCP connects your AI client directly to the world's largest NFT marketplace, giving you real-time market intelligence and asset tracking across multiple blockchains. Use it to analyze collection statistics, track floor prices, pull full metadata for any digital collectible, and monitor wallet balances instantly. It's your single source for deep-diving into Web3 assets without leaving your chat interface.

F Quality Score 3.6/100

nft | marketplace | digital-assets | crypto-collectibles | floor-price | web3

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.

OpenSea MCP

35 tools available
Cloud-hosted on Vinkius

Need to understand what's happening in the NFT market? This MCP gives your AI agent direct access to OpenSea's entire data set. Instead of manually jumping between tabs, dashboards, and multiple chain explorers just to figure out a collection's value or an asset's history, you simply ask your client. You can pull current trends using `get_trending_collections` or analyze specific collections by listing all traits with `get_collection_traits`. Want to know the true rarity of a piece? Use `get_nft_metadata` to retrieve every detail and link attached to an NFT. It also keeps track of wallets, letting you check account profiles and fungible token balances using `get_account` and `get_token_balances`. By connecting this MCP through Vinkius, your AI client becomes a dedicated market analyst, giving you data on ownership records, volume stats, and current listings across chains like Ethereum or Polygon. It transforms the overwhelming process of research into simple conversation.

Core Capabilities

01 — Analyze collection performance metrics

You retrieve comprehensive statistics for a whole collection, including total volume and floor price.

02 — Inspect individual NFT data

You pull detailed metadata, ownership history, and rarity traits for any specific digital asset.

03 — Track wallet holdings

You check a wallet's profile name and current balances for both fungible tokens and NFTs it owns.

04 — Monitor marketplace listings and offers

You list all active items in a collection or find the highest offer available on a single NFT.

05 — View market trends across chains

You query which collections are currently trending and track data across multiple supported blockchains.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP on Vinkius and provide your required OpenSea API Key.

02 Your AI client connects to the catalog and recognizes the available NFT tools.

03 You prompt your agent with a specific request, like 'Show me the top 5 trending collections,' and receive structured data instantly.

The bottom line is you get market-grade intelligence on digital collectibles without leaving the chat window or needing to manage complex API calls yourself.

Built For

This MCP is essential for Web3 developers, professional NFT traders, and data analysts who need real-time asset information. It solves the painful problem of having to open multiple browser tabs—one for stats, one for metadata, one for wallet balances—just to compile a single market report.

NFT Trader

You monitor floor prices and volume trends across competing collections instantly by using the MCP.

Web3 Developer

You use the MCP in your code editor to pull specific NFT metadata or contract ownership details for immediate project integration.

Market Researcher

You automate the collection of statistics and trait distributions across multiple collections for comparative market analysis.

What Changes When You Connect

01 You get instant access to collection statistics; instead of visiting multiple dashboards, you use `get_collection_stats` to pull total volume, floor price, and sales data in one query.

02 Deep-dive into any asset using `get_nft_metadata`. You retrieve all the critical details—image links, external documentation, rarity scores—without copy-pasting from web pages.

03 Check entire wallet portfolios with tools like `get_nfts_by_account` and `get_token_balances`. Your agent compiles a full picture of assets held by an address in one go.

04 Track market movement using `get_trending_collections` to see what's hot right now, or use `list_offers_collection_all` to gauge demand for specific groups of items.

05 You can analyze supply and rarity distribution by listing collection traits with `get_collection_traits`, helping you understand the market landscape far better than manual browsing.

Real-World Applications

Analyzing a competitor's portfolio

A client wants to know everything about a rival collector's holdings. They ask their agent to run `get_nfts_by_account` and also check the wallet's liquid assets using `get_token_balances`. The AI compiles a comprehensive report on both their NFTs and their available cash, all in one response.

Valuing a new collection launch

A developer needs to understand if a planned NFT drop is viable. They use the MCP to call `get_collection_stats` for similar existing collections and then run `get_collection_traits` to model potential trait scarcity, giving them immediate data points for their pitch.

Following up on a major transaction

You completed a swap or listing creation. Instead of waiting hours for confirmation, you immediately use `get_transaction_receipt` to check the status and confirm if the funds moved correctly or if the action failed.

Researching asset ownership lineage

A collector wants to know who originally owned a specific piece of art. They run `get_nft_owners` on the item's contract address and get a full, verifiable history of every wallet that has ever held it.

Patterns to Avoid

Trying to search for everything at once

✗ AVOID

Asking your agent: 'Tell me about the market. Give me stats on Bored Apes, and also what's trending, but also check out wallet 0x123.' This results in a disorganized mess of data fragments.

✓ INSTEAD

Break it down into specific tasks. First, ask for `get_trending_collections` to see market buzz. Then, run `get_collection_stats` only on the collection name you care about. Finally, check the wallet using `get_account`. Focus your prompts.

Missing the specific asset detail

✗ AVOID

If you just ask 'Tell me about a rare NFT,' the AI gives you general information but can't pull the specific rarity or metadata for one piece.

✓ INSTEAD

Always provide the unique identifier. Use `get_nft` and reference the specific token ID to get accurate metadata, traits, and ownership data.

Confusing listing details with collection rules

✗ AVOID

If you ask about a collection's general rule set but need to know if a specific item is for sale, you'll waste time.

✓ INSTEAD

First, use `get_collection_traits` to understand the possibilities. If you want to see what's available right now, run `list_listings_collection_all`. This separates general knowledge from active market data.

The Right Fit

Use this MCP if your workflow requires real-time, structured data about digital collectibles—specifically price floors, ownership records, or collection statistics. You need to know *what* the assets are worth and *who* owns them right now. Don't use it if you need legal contract analysis, complex financial modeling that involves external company data, or historical market sentiment (though `list_events` helps with raw history). If your goal is simply general news about Web3 trends without specific asset metrics, a simple search engine might suffice. But for any task requiring verifiable numbers like 'What's the lowest listing price?' (`get_best_listing_nft`) or 'How many people own this trait?' (`get_collection_traits`), this MCP is your single source of truth.

Market research used to be a graveyard of browser tabs.

Today, analyzing an NFT collection means opening OpenSea. You click on the collection page for stats; then you open another tab to check ownership via `get_collection_holders`. If you want the metadata, you copy-paste a token ID into a third tool just to get the image links and traits. It's endless clicking, constant switching between dashboards, and painfully manual data compilation.

With this MCP, your AI agent handles the entire process in one chat thread. You ask for a full market breakdown—stats, ownership, and rarity details—and it compiles all that information using tools like `get_collection_stats` and `get_nft_metadata`. You get the data instantly; no more tab switching.

OpenSea MCP Delivers Precise Asset Ownership Data

You don't have to manually query every type of asset. Instead of running separate queries for general holdings, you use `get_nfts_by_account` and then follow up with `get_token_balances`—all linked by the same wallet address. The system automatically connects those two data points for a complete picture.

It’s simple: You tell your agent what question to answer about the market, and it uses all the right tools—like `get_nft_owners`, `get_best_offer_nft`, or `list_offers_collection_all`—to deliver one definitive answer. The data is always current.

OpenSea MCP with 35 Tools

These tools let you query every aspect of the OpenSea marketplace—from listing new items to analyzing collection-wide statistics.

#	TOOL	DESCRIPTION
01	`cancel_order`	Removes an NFT listing from the marketplace before it sells.
02	`generate_listing_fulfillment`	Retrieves data needed to purchase a specific NFT listing.
03	`generate_offer_fulfillment`	Gathers the necessary information to accept an existing buyer's offer on an NFT.
04	`get_account`	Provides a summary profile and basic details for any given OpenSea account address.
05	`get_best_listing_nft`	Finds the absolute lowest price available listing for a specific NFT.
06	`get_best_offer_nft`	Identifies the highest current offer placed on a specific NFT.
07	`get_chains`	Lists every blockchain network that OpenSea supports for data queries.
08	`get_collection_holders`	Returns a list of wallet addresses that currently own items belonging to a specific collection.
09	`get_collection_stats`	Provides high-level statistics for an entire NFT collection, like total sales and volume.
10	`get_collection`	Retrieves detailed information about a specific NFT collection's rules or history.
11	`get_collection_traits`	Lists every possible trait that can exist within a given collection (e.g., 'Hat: Fedora').
12	`get_drop_by_slug`	Gets the minting stages, supply count, and details for a specific NFT drop.
13	`get_drops`	Lists featured or upcoming batches of NFTs available in drops.
14	`get_nft_metadata`	Fetches detailed information, including image URLs and external links, associated with an NFT.
15	`get_nft_owners`	Lists every single wallet address that has ever owned a specific NFT.

#	TOOL	DESCRIPTION
16	get_nft	Gathers the metadata, traits, ownership history, and rarity score for one unique NFT.
17	get_nfts_by_account	Returns all NFTs owned by a specific wallet address.
18	get_nfts_by_collection	Retrieves every NFT that belongs to a specified collection ID.
19	get_nfts_by_contract	Lists all NFTs associated with a specific smart contract address.
20	get_order	Retrieves the full details of an existing marketplace order or listing.
21	get_swap_quote	Generates a price estimate for exchanging one cryptocurrency or token for another.
22	get_token_balances	Checks the quantity of fungible tokens (like USDC) held by a specific wallet address.
23	get_token	Retrieves the metadata and details for a specific type of fungible token.
24	get_top_collections	Returns a list of collections that are currently performing well in sales volume or interest.
25	get_transaction_receipt	Verifies the status and outcome of a recent swap, listing creation, or fulfillment action.
26	get_trending_collections	Identifies which NFT collections are currently experiencing rapid growth in interest and sales.
27	list_collections	Provides a general list of available NFT collections on the platform.
28	list_events	Retrieves historical records of significant marketplace events and sales activity.
29	list_listings_collection_all	Fetches every active listing currently available for sale within a specific collection.
30	list_offers_collection_all	Retrieves all current offers placed on items within a specified collection.
31	post_criteria_offer	Creates an offer that applies across entire collections or based on specific traits.
32	post_listing	Publishes a brand new NFT item listing for sale in the marketplace.
33	post_offer	Places a direct, specific offer on one individual NFT asset.

#	TOOL	DESCRIPTION
34	post_swap_execute	Generates executable transaction data needed to swap crypto assets.
35	search	Runs a broad search across collections, accounts, NFTs, and tokens using keywords or filters.

See It in Action

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

U Show me the trending NFT collections on OpenSea right now.

I've fetched the trending collections. Currently, 'Pudgy Penguins' and 'Azuki' are seeing high activity. Would you like to see the specific stats for one of them?

U Get the floor price and volume stats for the 'boredapeyachtclub' collection.

The Bored Ape Yacht Club currently has a floor price of 12.5 ETH with a total volume of 1.2M ETH. There have been 45 sales in the last 24 hours.

U Check the profile and token balances for wallet address 0x123...

For wallet 0x123..., the profile name is 'CryptoCollector'. The account holds 2.5 ETH and 500 USDC. Would you like to see their NFT holdings as well?

Frequently Asked Questions

01 **How do I check the price history using OpenSea MCP?**

While we don't have a direct 'price history' tool, you can track trends by repeatedly calling `get_trending_collections` and reviewing the volume data over time. This allows your agent to build a rough performance graph for analysis.

02 **Can I check if an NFT is listed using OpenSea MCP?**

Yes, you can list all active items in a collection by calling `list_listings_collection_all` or check specific listings by running `get_order` and referencing the listing ID.

03 **What is the difference between `get_nft` and `get_nft_metadata` using OpenSea MCP?**

`get_nft` gives you a high-level summary, including traits and rarity. `get_nft_metadata` provides the deep dive—the actual image URLs, external links, and detailed descriptive text for that specific piece.

04 **How do I find out if an offer is available on OpenSea MCP?**

You can check two ways: use `list_offers_collection_all` to see all offers in a collection, or use `get_best_offer_nft` by providing the specific NFT ID you want to know about.

05 **Does OpenSea MCP support multiple blockchains?**

Yes. You can run `get_chains` to see every supported blockchain network, ensuring your analysis is comprehensive and not limited to a single chain.

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 `"opensea": { "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

OpenSea 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 OpenSea. 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	OpenSea MCP
Server ID	`019e38cf-0986-7153-8ff2-34787d3d5ea5`
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/opensea.