MCP SERVER | NO CODE | CLOUD HOSTED

Kibana MCP

Manage your entire observability stack via chat.

Kibana connects your observability stack directly to any AI agent. Use this MCP to manage Kibana spaces, discover dashboards, search index patterns, and automate configuration changes across your Elastic Stack. It lets you control everything from creating roles to copying saved objects without leaving the chat window.

A+ Quality Score 98.33/100

elastic-stack | dashboards | observability | data-visualization | kibana-api

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

SCAN TO VIEW

The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect 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.

Kibana MCP

55 tools available
Cloud-hosted on Vinkius

Managing a large-scale monitoring platform like Kibana usually means jumping between dozens of tabs just to check settings or move visualizations. This MCP handles that complexity for you, letting your agent interact with your entire Elastic Stack through natural language commands. You can list and inspect spaces, find specific index patterns, or copy saved objects directly into different environments. Need to change permissions? Use the tools here to create, update, or delete roles and even provision new agents. It's about controlling your data infrastructure from a single chat window. Vinkius makes it simple: connect once with any MCP-compatible client, and you get instant access to all these critical management functions.

Core Capabilities

01 — Manage Kibana Spaces

List, create, or delete entire workspaces (spaces) to organize dashboards for different teams.

02 — Control Saved Objects

Create, read, update, and delete saved items like index patterns and visualizations across your instance.

03 — Automate Role Permissions

Manage user roles by creating or updating specific Kibana permissions to control who sees what data.

04 — Maintain Data Consistency

Copy saved objects between different spaces, ensuring your monitoring dashboards look the same in staging and production.

05 — Audit System Components

Retrieve full metadata for any object—like a data view or an agent policy—so you can understand its current configuration.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your specific Kibana URL along with a valid API Key.

02 Connect the credentials to your AI client, giving your agent access to manage your Elastic environments.

03 Tell your agent exactly what you need done—for example, 'Find all dashboards related to networking in the Engineering-Logs space.' — and get the results.

The bottom line is that it turns complex, multi-step UI navigation into a single conversational request.

Built For

Anyone who spends too much time clicking through dashboards or manually copying configuration files needs this. It's for the Ops Engineer tired of logging into Kibana just to list spaces, and the Data Analyst who struggles to find a specific visualization across massive instances.

DevOps/SRE

You use this MCP to quickly audit space configurations or move dashboards between staging and production environments with simple commands.

Platform Engineer

You automate the provisioning of team-specific spaces, default saved objects, and user roles without writing boilerplate API scripts.

Data Analyst

You search for specific visualizations or index patterns across massive Kibana instances using natural language queries to find exactly what you need.

What Changes When You Connect

01 Instead of manually navigating to the 'list spaces' section, you simply ask your agent for a list of all available Kibana workspaces. This lets you immediately see every environment you manage, like finding all team-specific areas.

02 When you need consistency across environments, use `copy_saved_objects`. You can copy dashboards or index patterns between staging and production spaces with one command, guaranteeing that the view logic remains identical.

03 If a dashboard is broken because of outdated permissions, you don't have to find the right admin role. The tools let you `create_or_update_role` directly, fixing access control instantly through conversation.

04 Need to understand why something isn't showing up? You can use `get_saved_object` or `find_saved_objects` to pull all metadata on an index pattern, telling you its exact configuration and last updated time without clicking anything.

05 Setting up a new alert used to be tedious. Now, the agent handles it; just tell it what metrics to watch, and use `create_rule` or `update_rule` to build and fine-tune alerts automatically.

Real-World Applications

Migrating a Dashboard from Test to Prod

A SRE needs to move the 'API Latency Overview' dashboard from their testing space to the main production monitoring space. They ask the agent, and it runs `copy_saved_objects`, instantly making sure the dashboard is available in the correct location without manual clicking or file transfer.

Auditing User Permissions for Compliance

A Platform Engineer must verify that only senior team members can see certain logs. They use `list_roles` to see all roles, and then ask the agent to run `get_role` on 'Security-Admins' to confirm they have access to the necessary data views.

Fixing a Missing Index Pattern

A Data Analyst realizes that the new log source isn't appearing. They ask the agent to `find_saved_objects` using keywords like 'new service logs', and it points them to the exact index pattern they need, saving hours of searching.

Scaling Up Observability Infrastructure

A DevOps lead needs to set up a dedicated workspace for a new product team. They instruct the agent to `create_space` and then immediately run `list_connectors` to ensure all necessary data sources are available in that fresh environment.

Patterns to Avoid

Trying to fix permissions manually

✗ AVOID

A user gets the error 'Permission Denied' when viewing a dashboard and spends 30 minutes clicking through settings menus trying to figure out which role needs updating.

✓ INSTEAD

Instead, ask your agent to `get_role` details. If you find the right role, use `create_or_update_role` or `update_saved_object` to apply the necessary changes directly.

Manually searching for old data views

✗ AVOID

A team member needs a specific visualization but can't remember its name, forcing them to manually check every single data view in the list.

✓ INSTEAD

Use `find_saved_objects` and give your agent a natural language description of what you need. It searches the whole instance for you.

Rebuilding dashboards from scratch

✗ AVOID

A team member loses access to an old, complex dashboard and spends days trying to recreate all visualizations using raw data streams.

✓ INSTEAD

If they can't find the object, ask the agent to `get_saved_object` by ID or name. If it exists but is broken, use `update_saved_object` with the latest configuration details.

The Right Fit

Use this MCP if your workflow involves managing multiple, interdependent components within Kibana: spaces, roles, saved objects, and alerting rules. This tool excels when you need to perform auditing actions (like `list_roles` or `get_space`) or mass configuration changes (`bulk_create_saved_objects`). Don't use it if your goal is purely exploratory data analysis—if you just want to run a query on live logs, that's better handled by the Kibana UI itself.

However, if the problem is *managing* the infrastructure that powers the visualization (e.g., setting up access control or copying dashboards), this MCP is essential because it exposes all those administrative tools via conversation.

The Daily Pain of Observability Stack Management

Right now, managing your observability stack means clicking through a labyrinth of UI menus. You have to manually navigate between spaces, check roles in one tab, and then go back to another section just to list the saved objects you need. If you want to move a dashboard from staging to production, you're copying data in one place and updating permissions in three different tabs.

With this MCP, you treat your entire Kibana setup like a single system accessible through chat. You tell the agent what needs to happen—for example, 'Copy Dashboard X to Space Y.' The tool handles all the underlying steps: checking if the space exists, validating the object ID, and performing the copy operation—all without you touching a button.

Control Your Entire Stack with Kibana MCP

You eliminate redundant administrative tasks like checking if an index pattern is already defined or confirming which roles have viewing access. You don't have to remember the exact endpoint URL for every single object type; you just describe what you need done.

The difference now is control. Instead of being limited by a graphical user interface, your agent gives you programmatic power over every aspect of your data visualization layer.

Kibana MCP: 55 Tools for Observability Management

These tools allow you to query, fetch, collect metrics, and administer every part of your Kibana instance using natural language commands.

#	TOOL	DESCRIPTION
01	add_case_comment	Adds a comment directly to an existing case record.
02	bulk_create_saved_objects	Creates multiple saved objects at once using a bulk operation.
03	bulk_get_saved_objects	Retrieves details for several saved objects simultaneously in one call.
04	bulk_update_saved_objects	Updates multiple existing saved objects efficiently using a bulk operation.
05	copy_saved_objects	Duplicates and copies saved objects from one Kibana space to another.
06	create_agent_policy	Generates a new policy that controls the behavior of an Elastic Agent.
07	create_case	Creates a brand-new case record within the system.
08	create_connector	Sets up and creates a new data connector source.
09	create_data_view	Generates a specific view of the raw data for analysis purposes.
10	create_enrollment_key	Creates a new key used to enroll agents into your system.
11	create_or_update_role	Defines or modifies user roles and permissions within Kibana.
12	create_rule	Creates a new alerting rule that triggers when metrics cross set thresholds.
13	create_runtime_field	Adds or modifies a field in a data view without changing the original index structure.
14	create_saved_object	Generates and saves a single object like an index pattern or dashboard.
15	create_short_url	Creates a shortened, easily shareable URL for internal links.
16	create_space	Establishes an entirely new, isolated Kibana workspace (space).
17	delete_cases	Removes multiple case records from the system.
18	delete_connector	Deletes an existing data connector source.

#	TOOL	DESCRIPTION
19	delete_data_view	Permanently removes a specific data view from use.
20	delete_role	Removes a defined user role and all associated permissions.
21	delete_rule	Deactivates or removes an alerting rule that monitors metrics.
22	delete_saved_object	Permanently deletes a saved object, such as an index pattern or visualization.
23	delete_short_url	Removes a previously created short URL.
24	delete_space	Deletes an entire Kibana workspace, removing all contained dashboards and objects.
25	disable_rule	Turns off a specific alerting rule without deleting its configuration.
26	enable_rule	Restores an alerting rule, allowing it to start monitoring metrics again.
27	execute_connector	Triggers a manual run of a configured data connector action.
28	export_saved_objects	Generates export files containing sets of saved objects for archival or migration.
29	find_rules	Searches the system to list all existing alerting rules and their status.
30	find_saved_objects	Searches across your instance for specific saved objects using natural language queries.
31	get_agent	Retrieves detailed information about a specific Elastic Agent installation.
32	get_case	Fetches all details for an individual case record.
33	get_connector	Retrieves detailed metadata about a specific data connector source.
34	get_data_view	Fetches the full configuration and definition of a particular data view.
35	get_role	Retrieves all details about a specific Kibana user role, including permissions.
36	get_rule	Fetches the current state and definition of an alerting rule.
37	get_saved_object	Retrieves all metadata for a single, specified saved object type (like a dashboard or visualization).
38	get_short_url	Gets the full details and destination URL of a short link.
39	get_space	Retrieves all configuration details for a specific Kibana workspace.

#	TOOL	DESCRIPTION
40	import_saved_objects	Imports multiple saved objects into the current space from an uploaded file.
41	list_agent_policies	Lists all available agent policies that can be applied to your endpoints.
42	list_agents	Provides a list of every Elastic Agent currently reporting into the system.
43	list_connectors	Retrieves a complete catalog and status report of all connected data sources.
44	list_data_views	Lists every available data view defined within your Kibana instance.
45	list_enrollment_keys	Provides a list of all active enrollment keys for agent setup.
46	list_roles	Lists every defined user role and their default permissions in Kibana.
47	list_spaces	Gets a list of all available, top-level Kibana workspaces across the instance.
48	resolve_import_errors	Checks and reports on any errors that occurred during a bulk import process.
49	search_cases	Searches the case management system using keywords or filters to find relevant records.
50	unenroll_agent	Removes a specific Elastic Agent from monitoring and reporting within the stack.
51	update_cases	Modifies details for existing case records, such as changing status or adding notes.
52	update_data_view	Makes modifications to an existing data view's definition and mapping settings.
53	update_rule	Modifies the parameters, thresholds, or conditions of an active alerting rule.
54	update_saved_object	Updates the configuration of any saved object, such as changing a dashboard's source data.
55	update_space	Makes general modifications to an existing Kibana workspace's settings.

See It in Action

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

U List all available Kibana spaces.

I've retrieved the spaces. You have 3 spaces: 'Default', 'Engineering-Logs' (ID: eng-logs), and 'Security-Ops' (ID: sec-ops).

U Find all dashboards related to 'Network Traffic' in the 'Default' space.

Searching... I found 2 dashboards: 'Global Network Traffic Overview' (ID: net-01) and 'VPC Flow Logs Analysis' (ID: vpc-02). Would you like the details for one of them?

U Get the details for the saved object of type 'index-pattern' with ID 'logs-*'.

Fetching object... The index pattern 'logs-*' is configured with the time field '@timestamp' and contains 45 mapped fields. It was last updated on 2023-10-25.

Frequently Asked Questions

01 **How do I list all available Kibana spaces using the Kibana MCP?**

You use `list_spaces` to get a complete catalog of every workspace. This command gives you an overview of your entire observability setup, helping you know exactly where to look for specific dashboards.

02 **Can I update my user roles with the Kibana MCP?**

Yes, you can modify permissions using `create_or_update_role`. This tool allows you to define or change what users and groups are allowed to view across your data views.

03 **What is the best way to move a dashboard between environments?**

Use the `copy_saved_objects` tool. It copies saved objects, ensuring that dashboards maintain their full configuration when moving from a test space to production.

04 **How do I find specific index patterns with the Kibana MCP?**

You use `find_saved_objects`. Simply describe what you're looking for, and the agent searches across your entire instance catalog for matching saved objects.

05 **Does the Kibana MCP help me manage alerting rules?**

Absolutely. You can find all existing alerts with `find_rules`, or modify them using `update_rule` if a threshold needs adjusting, ensuring your system always monitors critical metrics.

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 `"kibana": { "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

Kibana 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 Kibana. 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	Kibana MCP
Server ID	`019e5d2a-ced9-72ea-a285-9f76c008effb`
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/kibana.