MCP SERVER | NO CODE | CLOUD HOSTED

Baseten MCP for AI Agents

Orchestrate machine learning model deployments and predictions

Baseten connects your AI agents directly to your machine learning infrastructure. Your agent can now manage entire model lifecycles—from listing deployed models to running real-time predictions on GPU weights and auditing sensitive workspace secrets.

A+ Quality Score 100/100

model-deployment | inference-api | serverless-ml | model-scaling | mlops

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

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.

Baseten MCP

6 tools available
Cloud-hosted on Vinkius

This MCP lets you treat your AI client like a full Machine Learning Operator. Instead of jumping through dashboards or writing complex scripts, your agent handles the whole process conversationally. You can ask it to list every model currently managed by Baseten, check the status of specific deployments, and even run direct predictions using tensor inputs. It's all about keeping your AI workflow contained, whether you're checking secrets or running inference on a new payload.

It gives you ML-Ops control right inside your chat window. When combined with Vinkius, you get access to this functionality alongside thousands of other services, letting your agent act as the single operational hub for your entire stack.

Core Capabilities

01 — List all deployed models

See a comprehensive list of every ML model currently managed within your Baseten account.

02 — Retrieve specific model details

Get full configuration information for any individual model ID you specify.

03 — Run serverless predictions

Execute real-time, low-latency inference by feeding tensor shapes or JSON directly into a deployed model instance.

04 — Audit active deployment states

List and inspect the current replica counts and autoscaling configurations for specific models.

05 — Check workspace secrets

Enumerate all active environment variables and secrets stored securely within your isolated ML orchestration space.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP on Vinkius and provide your Baseten API key.

02 Give your AI agent a command, like 'What models do we have?'

03 Your agent runs the necessary tool calls and responds with structured data, allowing you to take immediate action.

The bottom line is that your AI client becomes an integrated ML workflow toolset for Baseten.

Built For

This MCP solves the problem of context switching. It's built for technical people who spend too much time jumping between cloud consoles, local notebooks, and terminal windows just to check a model status or run a test payload.

ML Engineer

You use this MCP to execute immediate test payloads against deployed models without having to spin up local Python environments first.

DevOps/SRE

You audit running deployment resources, verify replica states, and check autoscaling configurations reliably from your core IDE interface.

AI Researcher

You inspect version schemas and manage complex inference pipeline architectures quickly to validate research hypotheses in production-like environments.

What Changes When You Connect

01 Run live inference tests immediately. Use the `predict` tool to test payloads against deployed models without ever leaving your agent interface.

02 Keep track of infrastructure status. The MCP lets you list active deployments and check replica states, so you always know if your model is running correctly.

03 Manage complex resources in one place. You can view all managed models using `list_models` and audit their full configurations without switching tabs.

04 Maintain security visibility. Use the `list_secrets` tool to confirm that critical environment variables are provisioned securely, without exposing plaintext values.

05 Simplify troubleshooting. Instead of digging through logs, you get direct access to deployment details via `get_deployment`, making root cause analysis faster.

Real-World Applications

Verifying model readiness before launch

An ML Engineer needs to confirm if a new version of the Defect-Detector-V2 works. They use their agent to check all active deployments via `list_deployments` and then run a small test payload using `predict`, getting immediate confirmation that the inference is stable.

Auditing infrastructure compliance

A DevOps engineer needs proof of secure credentials. They ask their agent to list secrets, verifying that the required API keys are present and correctly isolated within the workspace using `list_secrets`.

Debugging unexpected prediction failures

An AI Researcher notices performance dips. Instead of guessing, they use the agent to pull explicit deployment details via `get_deployment`, identifying if scaling parameters or version mismatches are causing the issue.

Onboarding a new team member quickly

A manager needs an overview of all assets. They ask their agent to list all managed models using `list_models` and get basic details on each one via `get_model`, providing a complete inventory summary.

Patterns to Avoid

Trying to manually copy configs

✗ AVOID

A user copies model IDs, deployment names, and secret keys into a local spreadsheet just to verify them later. This process is slow, error-prone, and provides zero real-time validation.

✓ INSTEAD

Use the agent to list models with `list_models` and then use `get_model` to get precise, structured configuration data for any specific model ID you need.

Running predictions on stale code

✗ AVOID

A developer runs a prediction using an old or unverified dataset payload because they didn't know the current deployment status.

✓ INSTEAD

Always check the current operational state first. Use `list_deployments` to ensure you target the latest, active inference bounds before attempting any predictions with `predict`.

Ignoring environment secrets

✗ AVOID

A new team member assumes all necessary API keys are available and starts coding without checking the secured environment.

✓ INSTEAD

First, use `list_secrets`. This confirms that your agent can see which secure credentials are provisioned in the workspace before you write any code that relies on them.

The Right Fit

Use this MCP if your workflow requires constant interaction with a deployed ML model's lifecycle. Specifically, if you need to check replica states (`list_deployments`), run immediate inference predictions (`predict`), or audit credentials (`list_secrets`), this is the right tool. Don't use it if all you need is simple data retrieval (like fetching text from a database); for that, look at generic data connector tools. If your goal is merely to write model code without testing it first, you might need a dedicated local IDE plugin instead of an MCP.

Baseten MCP: Managing ML Model Deployments with AI Agents

Right now, checking on your machine learning models is a nightmare. You're clicking between the cloud console dashboard to see if the service scaled correctly, then switching to a separate terminal window just to run a test payload against an endpoint, and finally opening a third tab to check environment variables because you forgot which API key was active. It’s slow, it requires too many hands-on clicks, and it’s impossible to audit everything in one place.

With this MCP, your agent handles the whole sequence. You tell it what needs checking—say, 'Give me status on Model X'—and it automatically pulls up deployment details, confirms the model config, and can even run a sample prediction. The result is structured, actionable data handed back to you in plain conversation.

Baseten MCP: Auditing ML Inference Infrastructure with AI Agents

Previously, verifying the operational health of an inference pipeline meant manually checking scaling rules and replica counts through multiple resource monitoring pages. If a key was missing, you had to navigate deep into the security settings just to confirm its existence.

Now, your agent manages this complexity. You can ask it to list deployments and get the specific details for any running instance in seconds. This capability moves infrastructure auditing from a half-day chore to a two-line chat command.

6 Tools in the Baseten MCP for Machine Learning Operations

These tools allow you to list models, check deployment status, run predictions, and manage sensitive secrets directly through conversation.

#	TOOL	DESCRIPTION
01	`list_models`	Retrieves a list of all machine learning models managed within the Baseten account.
02	`get_model`	Fetches detailed configuration data for one specific Baseten model ID.
03	`predict`	Runs a serverless inference prediction by passing explicit tensor shapes or dictionaries to the deployed model weights.
04	`list_deployments`	Lists all active deployment instances associated with a specific machine learning model.
05	`get_deployment`	Retrieves detailed operational information for a single, running deployment instance.
06	`list_secrets`	Displays all environment secrets configured in the workspace without revealing their actual values.

See It in Action

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

U I need to know what models we have deployed right now.

Here's the current inventory of managed models:

- **Llama-3-70B-Instruct:** Fine-tuned for conversational chat. (ID: XYZ)
- **Defect-Detector-V2:** Computer vision model for quality control. (ID: ABC)

You can also check the deployment status for either of these models to see their current replica counts.

U Can you run a quick test prediction on the Defect Detector model?

Sending payload to Defect-Detector-V2...

The inference pipeline returned successfully. Here is the result mapping:

```json
{
"score": 0.98,
"label": "PASS"
}
```

Prediction complete. This was executed against the latest stable deployment version.

Frequently Asked Questions

01 **How does Baseten MCP help me manage multiple AI models?**

It centralizes your entire ML model inventory. Instead of logging into separate dashboards for each service, you can ask the agent to list all deployed models and check their statuses from one place.

02 **Can I use Baseten MCP to test my model predictions?**

Yes, that's a core function. You can run immediate, real-time inference tests by providing specific payloads directly to the deployed models without needing local code setup.

03 **What if I need to check sensitive API keys or secrets? Does Baseten MCP handle that?**

The MCP lets you list all active workspace secrets. It confirms which credentials are provisioned and accessible for your models without ever showing the actual plaintext values, keeping everything secure.

04 **Does Baseten MCP help DevOps teams audit my ML infrastructure?**

Absolutely. You can check detailed deployment information, including replica counts and autoscaling configurations, allowing you to verify that your production environment is running exactly as designed.

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 `"baseten": { "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

Baseten 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 Baseten. 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	Baseten MCP
Server ID	`019d7558-a9f9-70f4-aef5-95adbac62678`
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/baseten.