MCP SERVER | NO CODE | CLOUD HOSTED

Google Cloud Storage MCP

Manage files, audit permissions, delete data via conversation.

Google Cloud Storage MCP lets you manage your entire cloud storage infrastructure through natural language conversation. Use your AI agent to list buckets, inspect file metadata, audit security permissions (IAM/ACLs), and move data—all without navigating the GCP Console. It's full control over GCS objects and buckets, handled by simple commands.

A+ Quality Score 100/100

bucket-management | file-storage | data-archiving | object-metadata | cloud-ops | access-control

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/google-cloud-storage

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.

Google Cloud Storage MCP

12 tools available
Cloud-hosted on Vinkius

Managing large cloud storage projects usually means spending hours in complex consoles, clicking through nested menus just to check a file size or verify who has access. This MCP changes that. You connect your Google Cloud Storage project, and your AI agent becomes your dedicated administrator. Instead of running API calls manually, you simply ask natural language questions: 'Show me the status of all development buckets' or 'Who can read the user logs in this bucket?' The agent handles the underlying complexity, reading the metadata, checking security policies, and executing operations like uploading new content or copying objects between locations. It means your AI client doesn't just talk to storage; it acts like a knowledgeable cloud ops specialist. This is how Vinkius makes powerful infrastructure tools accessible directly through conversation.

Core Capabilities

01 — Discovering and listing buckets

See a complete list of all buckets in your project, along with their specific location and storage class details.

02 — Finding files within buckets

Browse for objects inside a bucket using prefixes (like folders) to filter down thousands of stored files quickly.

03 — Uploading new data

Transfer text-based content or artifacts directly into any specified bucket in the cloud.

04 — Moving and deleting objects

Copy files from one bucket to another, or permanently delete specific objects that are no longer needed.

05 — Checking security policies

Audit the Access Control Lists (ACLs) and Identity and Access Management (IAM) policies for both entire buckets and individual files.

06 — Retrieving system details

Get detailed information about the project's service accounts or list keys used for cross-cloud integrations.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/google-cloud-storage** — connect your AI agent in three steps.

01 First, subscribe to this MCP and provide your Google Cloud Project ID along with your OAuth credentials.

02 Next, complete the required secure authorization flow to grant access to your cloud data within the system.

03 Finally, start asking your AI agent questions in Claude, Cursor, or any compatible client. It executes commands like listing buckets or checking permissions on demand.

The bottom line is, you get full control over complex cloud storage operations without ever opening a web console.

Built For

This connector is essential for Cloud Engineers who spend too much time manually checking build artifacts. It's perfect for Data Scientists needing to verify file sizes or modification dates across large datasets, and Security Teams that need instant audits of bucket permissions.

Cloud Engineer

Checks if a specific log file or build artifact exists in a staging bucket without logging into the GCP console.

Data Scientist

Browses large datasets to verify file sizes and check modification dates using conversational queries.

Security Analyst

Audits bucket permissions and public access settings instantly by asking the agent to check ACLs or IAM policies.

What Changes When You Connect

01 Audit compliance instantly: Instead of navigating complex IAM and ACL settings, simply ask the agent to check who has read or write access to a specific bucket using list_bucket_acl. You get an immediate pass/fail report.

02 Stop manual file checking: Need to know if 'build-v3.zip' exists? Ask your AI client to use list_object_acl and get_object_metadata, getting the status instantly without opening the console.

03 Efficient data movement: Use copy_object or upload_object to move datasets between staging and production buckets in a single conversational turn. No more multi-step GUI transfers.

04 Deep security oversight: The agent lets you check project service accounts using get_project_service_account, ensuring that cross-cloud integrations are running with the correct permissions.

05 Massive time savings for ops teams: Combining list_buckets and list_objects allows your AI client to map out complex data architectures—from finding all assets in 'prod' to locating a specific log file deep within 'logs/2024/'.

06 Full lifecycle management: Easily control the full object life cycle, whether you need to delete_object obsolete files or copy_object them for archival purposes.

Real-World Applications

Verifying compliance before launch

A security team needs to ensure that no sensitive user data is publicly exposed. They ask the agent to check 'user-uploads-data' bucket permissions using list_bucket_acl and get_bucket_iam, confirming public access ('allUsers') is blocked across all objects.

Archiving old datasets

A data scientist needs to move a year's worth of raw logs from 'logs/2023/' to the cold storage archive. They instruct the agent to list_objects for that prefix and then execute copy_object, keeping the original file intact.

Troubleshooting missing assets

A cloud engineer can't find a specific build artifact. Instead of opening the console, they ask the agent to list_objects using a known prefix ('assets/images/') and get_object_metadata to verify the exact file name and size.

Setting up new pipelines

A developer needs to test data flow. They use upload_object to push a dummy configuration file into the staging bucket, then ask list_buckets to confirm the asset is visible and available for downstream processes.

Patterns to Avoid

Using generic search terms

✗ AVOID

A user just searches 'find my data' in the chat. The AI client gets confused because it doesn't know which bucket, prefix, or date range to check.

✓ INSTEAD

Always be specific. Use list_objects and provide a clear path: 'list objects in prod-assets-9302 that start with images/2024/' to get precise results.

Assuming permissions are correct

✗ AVOID

A team member assumes the marketing assets folder is private, but accidentally publishes sensitive content because they never verified access controls.

✓ INSTEAD

Before any major upload_object or copy_object action, run list_bucket_acl and get_bucket_iam to confirm that only authorized service accounts have write permissions.

Doing manual object checks

✗ AVOID

A user has 50 files to check for size or modification date. They open the console, click into every folder, and manually copy-paste data points.

✓ INSTEAD

Use get_object_metadata on a list of objects identified by list_objects. The agent pulls all the required metrics in one query.

The Right Fit

You should use this MCP if your workflow involves complex, multi-step data governance: checking who can access what, moving files between different environments (dev to staging), or auditing a large number of assets. You need it when the answer isn't 'yes/no', but requires reading metadata and security policies across multiple buckets.

You don't use this if you just need simple file transfers without checking permissions; for that, a basic sync tool might suffice. Also, if your only task is to write code based on local files—and never touch the cloud environment—you shouldn't use it. This MCP is specifically for controlling and observing data *in* Google Cloud Storage.

The Pain of Manual Console Navigation

Today, managing a major cloud storage project means opening the GCP console. You click into the 'Storage' tab, then navigate to your specific bucket name. To check permissions, you have to find the IAM section, and if that isn't enough, you drill down further into ACLs—it's a dozen clicks just to answer one simple question: 'Is this public?'

With this MCP, you talk to your agent. You ask, 'What are the access rules for all assets in the staging bucket?' and it returns the full policy immediately. It's like having an expert cloud architect sitting next to you, instantly giving answers instead of sending you through a maze of menus.

Managing Objects with Google Cloud Storage MCP

The manual steps that vanish include opening the console, clicking 'Objects', filtering by prefix, and then checking each file's metadata one by one. You never have to click through folders again just to see what data exists.

You now simply ask your agent to list all objects matching a pattern, and it provides the full list with sizes and dates in plain text. It changes cloud operations from tedious clicking into simple conversation.

Google Cloud Storage MCP with 12 Tools

Use these tools to perform deep cloud operations, including listing buckets, managing object lifecycles, checking access controls, and transferring files through natural language prompts.

#	TOOL	DESCRIPTION
01	copy_object	Copies a file from one location to another, either within the same bucket or across different buckets.
02	delete_object	Permanently removes a specified object (file) from its current bucket.
03	get_bucket_iam	Retrieves the full IAM policy that controls who has access to an entire bucket.
04	get_bucket_metadata	Gets key information, like creation date and location, for a specific bucket.
05	get_object_metadata	Retrieves detailed metadata (size, type, dates) for a single file inside a bucket.
06	get_project_service_account	Checks the assigned service account details used by the project's storage resources.
07	list_bucket_acl	Lists all current permissions and access rules applied to an entire bucket.
08	list_buckets	Retrieves a list of every single storage bucket available in your project.
09	list_hmac_keys	Lists the unique HMAC keys associated with a service account for integration purposes.
10	list_object_acl	Checks and lists all permissions applied only to one specific object (file).
11	list_objects	Finds and lists all files within a bucket that match a given prefix or folder path.
12	upload_object	Transfers a new file from your local environment into an empty or existing cloud bucket.

See It in Action

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

U List all buckets in my Google Cloud project.

Retrieving buckets... You have 4 buckets: 1. 'prod-assets-9302', 2. 'staging-logs-backup', 3. 'user-uploads-data', 4. 'temp-exports'. Would you like the metadata for the 'prod-assets' bucket?

U Find all files in bucket 'prod-assets' that start with 'images/2024/'.

Browsing objects... I found 15 objects with that prefix. Notable files include 'logo-v1.png', 'hero-background.jpg', and 'icon-set.svg'. Would you like to see the metadata for 'logo-v1.png'?

U Check who has access to the 'user-uploads-data' bucket.

Auditing bucket ACL... The 'user-uploads-data' bucket is restricted to internal service accounts and specific team members (admins). Public access ('allUsers') is currently NOT granted. Would you like the full IAM policy details?

Frequently Asked Questions

01 **How do I check permissions on a specific file using Google Cloud Storage MCP?**

Use list_object_acl to check the access rules for any single file. This is much faster than checking the entire bucket's policy if you only care about one item.

02 **Can I list all my buckets using Google Cloud Storage MCP?**

Yes, use list_buckets to retrieve a complete roster of every single bucket in your project. This is the starting point for any large-scale audit or inventory task.

03 **What tool do I use to move files from one storage location to another?**

Use copy_object. This function lets you transfer data between buckets, which is safer than deleting and re-uploading the file manually.

04 **Does Google Cloud Storage MCP handle metadata retrieval for multiple files?**

Yes, you can request get_object_metadata or list_objects to gather size and modification dates for many files at once through a single query.

05 **If I need to delete old logs, which tool should I use with Google Cloud Storage MCP?**

Use delete_object. Remember that this action is permanent, so always confirm the file name and bucket before confirming deletion via your agent.

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 `"google-cloud-storage": { "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

Google Cloud Storage 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 Google Cloud Storage. 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	Google Cloud Storage MCP
Server ID	`019d75a8-405b-7147-9eeb-1afd99e2ae94`
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/google-cloud-storage.