

MCP SERVER

NO CODE

CLOUD HOSTED

# Uscreen MCP

Manage membership billing and content details.

Uscreen connects your video membership platform data directly into your agent. You manage everything from customer details and subscription plans to content libraries and payment transactions without leaving your workflow. It gives your AI client full visibility into every aspect of your online streaming business.

**A+** Quality Score 100/100

video-streaming

membership-site

subscription-billing

course-hosting

content-monetization



# 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.

### 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.

### 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.

### 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**

Encryption at rest

**Ed25519**

PKI vault signatures

**24h TTL**

Ephemeral session keys

**V8 Isolate**

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.

### 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.

### 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.

### 06 — Honeypot Trap System

Phantom credentials are injected into isolated environments. If a honeypot 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.

**FinOps**

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

**Firewall & DLP**

PII redaction activity, sensitive data protection counters, and security event timeline.

**Agent Activity**

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

**Tool Health**

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

**Incident Log**

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

Get started at [cloud.vinkius.com](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Uscreen MCP

10 tools available

Cloud-hosted on Vinkius

Need to run an analysis on your digital course or streaming service? This MCP connects your Uscreen account, letting your agent interact with all the data that runs your membership site. Instead of logging in and clicking through multiple dashboards, you ask for specific information—like listing all customers or checking top revenue generators. Your AI client can pull together a list of transactions, cross-reference it with active subscription plans, and even look up content details across different collections. Everything is accessible via the Vinkius catalog, meaning once you connect your agent, you get full access to manage everything from product catalogs to payment records.

---

## Core Capabilities

### 01 — Track membership revenue

List all past and current payment transactions to calculate total revenue and identify high-value periods.

### 02 — Audit customer accounts

Retrieve detailed profiles for specific customers or list every user on the platform.

### 03 — Manage course content

Browse and retrieve details about individual videos, entire collections, and available courses.

### 04 — Analyze subscription models

View all current membership plans and product offerings used for billing and sales.

### 05 — Verify platform status

Confirm that the API connection to Uscreen is active and running correctly.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/uscreen](https://vinkius.com/mcp/uscreen) — connect your AI agent in three steps.

- 01 First, you connect your agent to Vinkius using your Uscreen credentials.
- 02 Next, you tell your AI client exactly what data you need—for example, 'List all customers who signed up last month.'
- 03 Your agent executes the necessary calls, pulls the raw customer and transaction data, and presents it back to you in a structured format.

The bottom line is that your AI client becomes an extension of your Uscreen backend, letting you access operational data without manual logins or API coding.

---

## Built For

This MCP is for the Digital Product Manager who manages membership sites and e-learning platforms. You're tired of jumping between billing dashboards, content management systems, and analytics tools just to answer a simple question about revenue or churn rates.

### E-Learning Platform Manager

Uses this MCP to list all available courses and check the status of membership plans so they can quickly update pricing details.

### Revenue Operations Analyst

Runs reports by listing transactions and checking customer profiles to identify patterns in churn or unexpected revenue drops.

### Content Strategist

Uses this MCP to list content and check collection details, ensuring all new videos are properly categorized and available for purchase.

---

## What Changes When You Connect

- 01 Stop manually tracking sales. Use `list_transactions` to pull all payment records in one go, giving you immediate visibility into monthly revenue streams.

- 
- 02** Keep your users organized. Instead of querying multiple databases for user info, use `list_customers` or `get_customer` to fetch complete profiles and activity history instantly.
- 
- 03** Audit your curriculum structure easily. You can `list_collections` and then `get_content` to confirm that every piece of educational material is correctly linked within its intended group.
- 
- 04** Keep billing accurate. `List_subscriptions` gives you a clean overview of all available membership plans, helping agents answer client questions on the spot.
- 
- 05** Maintain connectivity confidence. Running `check_uscreen_status` confirms your connection health before critical reports are run, preventing workflow interruptions.
- 

---

## Real-World Applications

### Identifying top-performing courses

The Content Strategist needs to know which videos drive the most sales. They ask their agent to `list_content` and cross-reference that data with `list_transactions` to pinpoint exactly which courses lead to high revenue.

### Investigating a specific customer's account

The support team needs details on a problematic user. They ask their agent to `get_customer`, which pulls all necessary profile data, and they can then check `list_transactions` for recent payment issues.

### Onboarding a new product tier

The Platform Manager needs to update pricing across the site. They use `list_products` to see what currently exists, and then use `list_subscriptions` to ensure the new membership plan is added correctly before going live.

### Generating monthly financial reports

The analyst must compile revenue figures. Instead of logging into the billing system, they ask their agent to `list_transactions`, getting a complete record that shows total revenue and average transaction size.

---

## Patterns to Avoid

---

### Trying to find content without knowing the collection name

#### X AVOID

A user might try to search for 'the latest videos' across all platforms, only finding fragmented results because they don't know the specific collection ID.

#### ✓ INSTEAD

First, use `list_collections` to see your high-level categories. Then, specify the exact group name when calling `get_collection` or `list_content` so you pull a clean, focused set of data.

---

### Confusing user listing with product inventory

#### X AVOID

A request like 'show all things' results in confusing mixed output that combines actual customers and purchasable items.

#### ✓ INSTEAD

Separate your queries. Use `list_customers` for people, and use `list_products` or `list_subscriptions` when you need to see what is available for purchase.

---

### Assuming transactions cover all revenue

#### X AVOID

The user only checks the transaction log (`list_transactions`) but misses manual sign-ups that don't generate immediate payments.

#### ✓ INSTEAD

Always cross-reference `list_customers` with `list_subscriptions`. This confirms who is registered and what their active plan is, giving a full picture beyond just payment records.

---

## The Right Fit

Use this MCP if your core business process involves selling structured digital content or memberships. You need to connect data points: 'Does the customer (`get_customer`) have an active subscription (`list_subscriptions`)? And if so, what content (`list_content`) are they allowed access to?' This is for operational oversight and reporting. Don't use this if your primary need is marketing analytics or deep behavioral psychology; that requires tools focused on user interaction tracking outside of transaction records. If you just need a simple form submission system, you might prefer a dedicated contact management tool instead.

---

---

## The Billing and Content Nightmare

Today, figuring out if your new pricing model works involves logging into the billing dashboard to check `list_subscriptions`, then opening the content library to verify video access permissions, and finally pulling a report of transactions—all while copy-pasting IDs between separate systems. It's slow, it's error-prone, and you spend more time on data retrieval than strategy.

With this MCP, your agent handles that entire sequence in one prompt. You ask for the full picture—for example, 'Show me all customers who signed up last week but haven't paid.' The result is a clean list of names and their specific account details, instantly solving complex cross-system checks.

---

## Uscreen MCP Gives You Total Operational Control

Manual tasks like listing all available products (`list_products`) or verifying if a specific collection is ready for launch are tedious and interrupt your flow. Every time you manually check these details, you risk missing an out-of-sync item.

Now, those checks happen instantly through natural language prompts. You don't just get data; you get the confidence that your entire membership ecosystem—from the product catalog to customer status—is fully visible and actionable right where you are working.

---

# Uscreen with 10 Tools

Use these tools to pull specific data points on customers, payments, content, and collections directly into your agent for analysis.

#	TOOL	DESCRIPTION
01	<code>check_uscreen_status</code>	Verifies the API connection status to ensure data can be retrieved successfully.
02	<code>get_collection</code>	Fetches specific details about a defined content collection.
03	<code>get_content</code>	Retrieves detailed information for a single video or educational course piece.
04	<code>get_customer</code>	Pulls specific details about one individual customer account.
05	<code>list_collections</code>	Returns a list of all existing content collections on the platform.
06	<code>list_content</code>	Lists every piece of video or educational content hosted on Uscreen.
07	<code>list_customers</code>	Retrieves a comprehensive list of all user accounts on the platform.
08	<code>list_products</code>	Returns an inventory of all products available for purchase or subscription.
09	<code>list_subscriptions</code>	Provides a full listing and overview of active membership plans and their pricing structure.
10	<code>list_transactions</code>	Lists all payment records, allowing you to track revenue over specific time periods.

---

## See It in Action

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

### **U** List all my Uscreen customers.



You have 1,234 customers. Active: 987, Churned: 247. Top recent signup: 'Jane D.' (Premium plan, joined today).

### **U** Show all subscription plans.



3 plans: 'Basic' (.99/mo, 312 subscribers), 'Premium' (.99/mo, 489 subscribers), 'VIP' (.99/mo, 186 subscribers).

### **U** List all payment transactions.



247 transactions this month. Total revenue: ,450. Average transaction: .40. Top: Premium renewals (67%).

---

## Frequently Asked Questions

### **01** How do I check revenue using Uscreen MCP?

You use `list_transactions`. This tool pulls all payment records, letting your agent calculate total revenue, average transaction size, and identify the best-performing time periods.

### **02** Can I see a customer's full history with Uscreen MCP?

Yes, you can `get_customer`. This tool pulls detailed profiles that show everything about an individual user, including their subscription status and associated content access rights.

---

**03 What is the difference between list\_content and list\_collections in Uscreen MCP?**

list\_collections shows you the organizational containers (the folders), while list\_content gives you the actual, detailed information for every individual video or course within those collections.

---

**04 Does Uscreen MCP help me manage my product catalog?**

Yes. You use list\_products to see everything available for purchase and list\_subscriptions to view the various membership plans that power your sales structure.

---

**05 What if I need to check connectivity first with Uscreen MCP?**

Always start by calling check\_uscreen\_status. This quick call confirms the API connection is active, so you know your agent can successfully pull data before running long reports.

---

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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 `"uscreen": { "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

# Uscreen 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](https://cloud.vinkius.com) →

[vinkius.com](https://vinkius.com) · [support@vinkius.com](mailto: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 Uscreen. 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	Uscreen MCP
Server ID	019dd17f-ebe5-7311-b4a6-3daaf26a6641
Platform	Vinkius Cloud for AI Agents
Endpoint	<code>https://edge.vinkius.com/{token}/mcp</code>

### 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/uscreen](https://vinkius.com/mcp/uscreen).