MCP SERVER | NO CODE | CLOUD HOSTED

Statuspage (Atlassian) MCP

Manage all page access and roles via conversation.

Statuspage (Atlassian) connects your AI agent directly to your status page infrastructure. Manage all corporate announcements, user roles, and branding from a simple chat interface. You can list pages, update domains, control private access for specific clients, and handle permissions—all without logging into the Statuspage dashboard.

A+ Quality Score 98.33/100

incident-management | status-page | atlassian | uptime-monitoring | it-operations

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/statuspage-atlassian

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.

Statuspage (Atlassian) MCP

16 tools available
Cloud-hosted on Vinkius

When an incident hits, you don't want to jump between dashboards or remember complex API calls just to check who has access or what the page domain is. This MCP lets your AI agent monitor and manage your entire incident communication stack through natural conversation. You can list all status pages and pull detailed configurations for specific IDs using simple commands. Need to update a page name, change branding settings, or modify the primary domain? That's handled instantly. Furthermore, if you have private pages for select clients, you can query who has access or create new user accounts with granular controls over components and metrics. This capability is crucial because it lets your agent handle complex tasks like managing user roles and updating permissions across an entire organization. Vinkius brings all these tools together so that regardless of which AI client you use, the functionality stays in one place.

Core Capabilities

01 — List and view page details

Retrieves a list of all status pages and fetches full configuration details for any specific page ID.

02 — Modify page settings and branding

Allows you to update core elements like the page name, domain, or overall branding across your entire Statuspage setup.

03 — Manage user permissions and roles

Checks existing user access levels or updates specific roles for users within your organization's status infrastructure.

04 — Control private page access

Manages audience-specific users, including creating new accounts or deleting access rights for private client pages.

05 — Update embedded widgets and configurations

Adjusts how status information appears on external sites by managing embed widget settings.

One Click on Vinkius — From Prompt to Execution

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

01 First, subscribe to this MCP and provide your Statuspage API Key.

02 Next, simply instruct your AI client. For example: 'List all my production pages' or 'Update page abc123 domain'.

03 The agent sends the request to the platform and reports back the status of the action—success, failure, or data retrieved.

The bottom line is you manage complex infrastructure changes using plain English prompts instead of logging into multiple dedicated UIs.

Built For

This MCP is essential for Site Reliability Engineers (SREs) and DevOps teams who face critical incidents after hours. It's also perfect for Support Managers needing to onboard or restrict access for high-profile clients without involving the core IT team.

Site Reliability Engineer (SRE)

During an outage, they need to quickly check if a specific page is configured correctly or update its domain name instantly from their terminal chat.

DevOps Lead

They use it to manage the lifecycle of status pages, ensuring correct branding and proper user roles are set up before launch.

Technical Support Manager

When a high-value client needs access to a private status page for testing, they can create or modify that specific user's permissions without manual intervention.

What Changes When You Connect

01 Stop hunting through multiple dashboards. You can list all pages or get detailed configurations using `list_pages` or `get_page`, getting the data you need in one chat exchange.

02 Need to onboard a new client? Use `create_page_access_user` and then refine their access with `update_page_access_user`. It handles complex user management without UI clicks.

03 Change branding or domains fast. Updating core settings like page names or domains using `update_page` means you can respond to an incident change immediately.

04 Control privacy tightly. You don't have to manually verify permissions; use `get_user_permissions` and `list_page_access_users` to see who has access right now.

05 Keep your site consistent. If branding changes, you update the embed settings with `update_status_embed_config`, ensuring all external widgets look right.

Real-World Applications

Restricting client visibility after a sale

A support manager needs to revoke access for an old customer who shouldn't see the status page anymore. Instead of finding and deleting their profile, they simply tell their agent: 'Delete page access user XYZ.' The agent executes `delete_page_access_user`.

Changing a domain during a merger

The DevOps team needs to shift the main status page's hosted domain immediately. They instruct their agent to run 'Update page abc123 with new domain.' The agent executes `update_page` and confirms the change.

Auditing user access before a public release

A Product Owner needs to audit who has permission to edit sensitive pages. They ask their agent to run 'List all page access users' and then use `get_user_permissions` on key accounts.

Adding new metrics for a premium client

The technical team discovers a new metric needs tracking for a specific private page. They use the agent to run 'Update page access user metrics' to add the required data point without touching the main UI.

Patterns to Avoid

Over-relying on API documentation

✗ AVOID

The engineer spends 20 minutes cross-referencing which endpoint to use for changing a user's role versus updating the page name.

✓ INSTEAD

Just tell your agent what you need: 'Update the user permissions for admin.' The agent knows whether it needs to run `update_user_permissions` or something else, saving time and guesswork.

Forgetting granular access control

✗ AVOID

The team updates a page but forgets that certain components or metrics need to be restricted for specific users.

✓ INSTEAD

Don't just update the user; use `get_page_access_user_components` first. Then, if needed, run `update_page_access_user_components` to precisely control what they see.

Using general-purpose scripts

✗ AVOID

Writing a complex script just to list all pages and their current owners.

✓ INSTEAD

Just ask your agent: 'List page access users for all my status pages.' The combination of `list_pages` and `list_page_access_users` gets the full picture instantly.

The Right Fit

Use this MCP if your workflow requires managing multiple, interconnected settings on a status page—like controlling both user roles AND branding domains. It's perfect for DevOps teams who need to audit access or make emergency changes (e.g., updating the domain) without logging into the Statuspage UI. However, don't use it if you just need general information that doesn't change frequently; for example, simply reading a static page description is better handled by direct API calls outside of the MCP. If your goal is solely to manage user credentials and nothing else, you might only need tools related to access control. But generally, this MCP provides the necessary depth (managing components, metrics, roles) that single-function tools lack.

The Pain of Manual Status Page Management

Today, updating status pages feels like a bureaucratic nightmare. You open the dashboard to change a domain name, but then you realize you also need to check if three specific clients still have access rights. That means clicking through user management tabs, switching between roles, and copying/pasting IDs just to verify permissions.

With this MCP, that whole process disappears. Your agent handles the sequence of actions for you. You tell it: 'Change the domain AND confirm Client X's role.' The result is a single, confirmed action from your chat window.

Managing User Access with Statuspage (Atlassian)

The manual steps that disappear include checking individual user roles, verifying components for specific client groups, and updating metrics separately. Each action usually requires a separate login or API call.

Now, you get centralized control. You manage the entire lifecycle of page access—from initial creation via `create_page_access_user` to fine-grained updates using `update_page_access_user_components`. It’s one conversation, done.

Statuspage (Atlassian): 16 Tools

These tools let you perform every major administrative task on Statuspage—from listing all pages to managing complex user permissions and branding settings.

#	TOOL	DESCRIPTION
01	`create_page_access_user`	Adds a new user account to manage access on private status pages.
02	`delete_page_access_user_components`	Removes specific components associated with a page access user.
03	`delete_page_access_user_metrics`	Deletes performance metrics linked to a page access user account.
04	`delete_page_access_user`	Completely removes an existing page access user from the system.
05	`get_page_access_user_components`	Retrieves a list of components currently assigned to a specific page access user.
06	`get_page`	Fetches all the detailed configuration settings for a single, specified status page ID.
07	`get_status_embed_config`	Gets the current settings used to embed status updates on external websites.
08	`get_user_permissions`	Checks and reports what level of access a given user account possesses within Statuspage.
09	`list_page_access_users`	Retrieves a comprehensive list of all page access users currently set up on the system.
10	`list_pages`	Provides a simple list containing the names and IDs of every status page you own.
11	`update_page_access_user_components`	Adds or replaces component settings for an existing page access user account.
12	`update_page_access_user_metrics`	Adds or replaces metrics linked to a specific page access user's profile.
13	`update_page_access_user`	Modifies the core details and settings for an existing page access user.

#	TOOL	DESCRIPTION
14	update_page	Updates fundamental properties of a status page, such as its name or associated domain.
15	update_status_embed_config	Changes the configuration settings for how your status pages are embedded onto external websites.
16	update_user_permissions	Modifies a user's role and permissions across multiple parts of the Statuspage platform.

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 Statuspage pages.

I've retrieved your pages. You have access to: 'Main Production' (ID: abc123) and 'Internal Services' (ID: def456).

U Show me the details for page ID abc123.

Fetching details for page abc123... The page is named 'Main Production', hosted at status.example.com, and currently has branding enabled.

U Update the name of page abc123 to 'Global Operations Status'.

I have successfully updated the name of page abc123 to 'Global Operations Status'.

Frequently Asked Questions

01 **How do I list all my pages using Statuspage (Atlassian) MCP?**

You use the `list_pages` tool. This simply returns a list of names and IDs for every page linked to your account, letting you know what resources you can work with.

02 **Can I update user roles using Statuspage (Atlassian) MCP?**

Yes, you use the `update_user_permissions` tool. This allows you to modify a user's specific roles and permissions across the platform in natural language.

03 **What if I need to change the domain name of my status page?**

You must use the `update_page` tool. This function lets you update fundamental properties, including changing the page's associated domain or branding settings.

04 **How do I delete a client's access account with Statuspage (Atlassian) MCP?**

You use `delete_page_access_user`. If that user has components or metrics attached, you should first run the respective delete tools to clean up fully.

05 **Does Statuspage (Atlassian) MCP handle embed widget settings?**

Yes. You use `get_status_embed_config` and `update_status_embed_config` to check or modify how your status is displayed on external websites.

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 `"statuspage-atlassian": { "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

Statuspage (Atlassian) 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 Statuspage (Atlassian). 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	Statuspage (Atlassian) MCP
Server ID	`019e38f3-6f69-7060-8dfd-40faf1ec3e22`
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/statuspage-atlassian.