

M C P S E R V E R

N O C O D E

C L O U D H O S T E D

Bitbucket MCP for AI Agents

Manage your Git repositories and CI/CD pipelines from your favorite AI client.

Bitbucket MCP. Connect your Bitbucket Cloud account to any AI agent to manage repositories, pull requests, and pipelines. No more tab-switching to check build statuses or review code. It lets your agent handle the heavy lifting of repository oversight and issue tracking directly from your workspace.

A+ Quality Score 100/100

git

version-control

pull-requests

ci-cd

repository-management

code-review



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 — connect your AI agent in under 60 seconds.

Bitbucket MCP

10 tools available

Cloud-hosted on Vinkius

You're tired of jumping between your code editor and a dozen browser tabs just to see if a build passed or if a colleague approved your latest pull request. This Bitbucket MCP changes that by bringing your repository data directly into your AI's context. Instead of manual searching, you can just ask your agent to summarize the latest activity or find a specific commit from last week. It handles the tedious parts of repository management, like listing branches or pulling issue lists, so you can stay focused on writing code. Because Vinkius hosts this MCP, you can get it running in minutes and start using it across any of your preferred AI clients. It's about having the right information at your fingertips without the friction of constant context switching.

Core Capabilities

01 — View all repositories

See every project within your workspace at once.

02 — Fetch pull request details

Get the specific status and information for any individual pull request.

03 — Retrieve latest commits

Pull the recent history of changes for any given repository.

04 — Check CI/CD pipeline statuses

See if your builds are passing or failing without leaving your chat.

05 — List open issues

Get a list of all issues for a repository that has trackers enabled.

06 — Access user profiles

Get the profile information for the authenticated user.

One Click on Vinkius — From Prompt to Execution

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

- 01 Subscribe to the Bitbucket MCP on Vinkius.
- 02 Enter your Bitbucket Username and App Password.
- 03 Ask your AI client to list repos, check PRs, or monitor pipelines.

The bottom line is you get a direct line to your Bitbucket data without leaving your workflow.

Built For

This is for the engineer who spends half their day hunting for PR statuses or the DevOps lead who needs a quick pulse on build health across multiple projects.

Software Engineer

Checking PR statuses and pipeline health during a sprint without switching tabs.

DevOps Engineer

Monitoring build failures and repository activity across the whole workspace.

Team Lead

Getting a summary of recent commits and open issues for daily standups.

What Changes When You Connect

- 01 Stop tab switching by using `list_pull_requests` to see what needs review without leaving your editor.
- 02 Monitor build health instantly with `list_pipelines` to catch failed CI/CD runs the moment they happen.
- 03 Get a clear view of the backlog by using `list_issues` to pull repository tasks into your chat.

-
- | | |
|-----------|---|
| 04 | Navigate complex projects faster by using <code>list_branches</code> to see all active development paths. |
| <hr/> | |
| 05 | Speed up your standups by using <code>list_commits</code> to summarize recent changes for the team. |
| <hr/> | |
| 06 | Gain a full project overview by using <code>list_repositories</code> to see every repo in your workspace. |
-

Real-World Applications

Checking PR status

An engineer asks 'Are there any open PRs for the auth module?' and the agent uses `list_pull_requests` to give an instant summary.

Commit history

A developer wants to know who touched the API last and asks the agent to use `list_commits` to find the recent changes.

Build monitoring

A DevOps pro asks 'Did the last pipeline for the web app succeed?' and the agent uses `list_pipelines` to check the status.

Workspace audit

A manager wants to see every repo they own and asks the agent to use `list_repositories` to get a full list of projects.

Patterns to Avoid

Trying to manage secrets

✗ AVOID

Asking the agent to store or generate your Bitbucket passwords.

✓ INSTEAD

Use a dedicated App Password created in your Bitbucket settings to authenticate the MCP.

Expecting full code edits

✗ AVOID

Asking the agent to rewrite 500 lines of code inside a repository.

✓ INSTEAD

Use this MCP to find the right PR or commit, then open the code in your editor for actual editing.

Missing trackers

X AVOID

Trying to use `list_issues` on a repository that doesn't have trackers enabled.

✓ INSTEAD

Ensure your repository has the issue tracker enabled in your Bitbucket settings before asking the agent for a list.

The Right Fit

Use this if you need to interact with Bitbucket Cloud data like PRs, pipelines, and issues from your AI client. It's perfect for high-level oversight and status checking. Don't use it if you need to perform complex code refactoring or deep file system edits. For those tasks, you should stick to your native IDE tools or a dedicated coding agent that has direct file system access. This MCP is your bridge to Bitbucket's metadata, not a replacement for your local git commands or deep code manipulation.

Bitbucket MCP: Automate Git Repository Oversight and PR Tracking

Most developers spend a huge chunk of their day context switching. You're in your IDE, then you jump to the browser to check a pull request, then you flip over to the CI/CD dashboard to see if the build is green. It's a constant cycle of clicking tabs and copy-pasting status updates into Slack or standup notes.

This Bitbucket MCP puts that data where you're already working. You can ask your agent to pull the latest PR status or list the active branches directly. You get a unified view of your work without the constant friction of browser tabs.

Bitbucket MCP: Monitor CI/CD Pipelines and Issue Trackers

You can stop manually hunting for specific commits or digging through long lists of issues.

It turns your AI client into a command center for your entire development workflow. You get faster

Instead of searching, you just ask your agent to find the last five commits or list all open issues for a specific repository.

answers and a much smoother day.

Bitbucket 10-tool Git Repository Management

Use these tools to fetch PR data, monitor pipelines, and manage your Bitbucket workspace metadata via your AI agent.

#	TOOL	DESCRIPTION
01	get_pull_request	Get the specific details and status of a single pull request. It helps you see exactly what's happening in a specific code review.
02	list_commits	See the history of changes for a repository. Use this to quickly find where a specific bug might have been introduced.
03	get_repository	Pull the metadata and details for a specific project. This gives your agent the context it needs to understand the repo.
04	get_user_profile	Fetch the profile details of the authenticated user. It helps the agent know who it's working with.
05	list_branches	See all the active branches in a repository. This is great for navigating complex multi-branch workflows.
06	list_issues	Pull a list of all issues for a repository with trackers. Use this to see the current backlog of work.
07	list_pipelines	Get the status of all CI/CD pipelines. This lets you know if your builds are green or red instantly.
08	list_pull_requests	See all open and closed pull requests for a repo. It's the fastest way to see what's pending review.
09	list_repositories	See every repository within your workspace. This gives you a high-level view of all your projects.
10	list_workspaces	See all the workspaces you have access to. Use this to navigate between different team environments.

See It in Action

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

U What's the status of the latest build for the mobile app?



The latest pipeline for 'mobile-app' (Build #882) completed successfully in 4 minutes and 12 seconds.

U List all open pull requests for the 'frontend' repo.



I've found 3 open pull requests for 'frontend':

ID	TITLE	STATUS
42	Refactor Login Flow	Open
45	Update CSS Grid	Open
49	Fix Header Alignment	Open

U Show me the last 3 commits on the 'backend' project.



Here are the recent commits for 'backend':

- **Auth Fix:** Resolved a token expiration bug.
- **Update Docs:** Added new endpoints to the README.
- **Refactor:** Cleaned up the database connection logic.

Frequently Asked Questions

01 Does the Bitbucket MCP work with private repositories?

Yes, it connects to your Bitbucket Cloud account so it can access any repository you have permission to view.

02 Can I use the Bitbucket MCP with Cursor or Windsurf?

Yes, it works with any MCP-compatible client, including Claude, Cursor, and Windsurf.

03 Do I need to install anything on my machine?

You just need to connect your Bitbucket credentials through Vinkius and link it to your AI client.

04 Can this MCP help me with my CI/CD builds?

It allows your agent to check the status of your Bitbucket Pipelines so you can see if builds are passing or failing.

05 How do I get my Bitbucket credentials?

You'll need to create an App Password in your Bitbucket account settings to share with the MCP.

06 Can it see my issues?

It can list issues for any repository that has the issue tracker enabled in your workspace.

07 Can I check the status of my latest Bitbucket Pipeline?

Yes! Use the `list\_pipelines` tool with your Workspace and Repository slug. Your agent will fetch the latest build statuses, showing if they passed or failed.

08 How do I list all open pull requests for a repository?

Simply ask the agent to `list\_pull\_requests` and provide the Workspace and Repository slug. It will retrieve all current PRs awaiting review.

09 Does the integration allow merging code directly?

Currently, the toolset is focused on querying and monitoring (Read-Only). You can list and inspect PRs, commits, and branches, but destructive actions like merging or deleting branches must be done through the Bitbucket UI for safety.

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 <code>"bitbucket": { "url": "..." }</code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → 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

Bitbucket 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 Bitbucket. 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	July 2026
MCP Server	Bitbucket MCP
Server ID	019d755d-8a2b-7189-96d5-211526454ae3
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/bitbucket.