MCP Server Frequently Asked Questions (FAQ)

Answers to questions you may have about the Stack Overflow Internal MCP server.

Document generated 12/02/2025

PDF VERSION

Tags | **Integrations** | **MCP Server** | **FAQ** |

Applies to: Free Basic Business **Enterprise**

*This documentation is for **Stack Internal Enterprise**. Free, Basic, and Business users can access their documentation here. Find your plan.*

Overview

The Stack Overflow Internal MCP Server is an open-source protocol layer that allows AI agents—like IDE assistants and internal copilots—to securely interact with verified knowledge in Stack Overflow Internal sites. It enables contextual, real-time access to internal content via API while preserving enterprise-grade security and attribution, unlocking scalable AI use cases directly in developer workflows.

Security and compliance

Data access and privacy

Where does the MCP Server run—on Stack Overflow's infrastructure or within the customer's environment?
The MCP Server is hosted and managed by Stack Overflow as part of each customer's Stack Overflow Internal site. Every customer has a dedicated endpoint (for example, https://yourteam.stackenterprise.co/mcp) that connects securely to that tenant's Stack Overflow Internal data.

Does any company data ever leave our environment or get stored by Stack Overflow?
No. The MCP Server operates entirely within the customer's Stack Overflow Internal environment, which is hosted and managed by Stack Overflow. When the server retrieves data, it does so securely through the Stack Overflow Internal API and streams results to authorized agents in real time. No customer content, queries, or credentials are stored or transmitted outside the tenant's Stack Overflow Internal site.

What data does the MCP Server access from our Stack Overflow Internal site?
The MCP Server accesses only the content available through your Stack Overflow Internal API, such as questions, answers, articles, comments, and related metadata. It follows the same permissions and visibility rules already configured in your Stack Overflow Internal site—meaning agents can only retrieve or write to content their authenticated users can access.

How does the MCP Server authenticate to our Stack Overflow Internal API?
The MCP Server uses an OAuth 2.0 Authorization Code flow with PKCE to authenticate securely to the Stack Overflow Internal API (v3). The MCP Server acts as an OAuth client, obtaining a short-lived access token on behalf of the signed-in user. Each request to the Stack Overflow Internal API includes this token as a bearer credential, ensuring all access is

scoped to the user's existing Stack Overflow Internal permissions. Tokens are tied to individual Stack Overflow Internal user IDs and expire automatically based on standard OAuth lifetimes.

Does Stack Overflow ever have visibility into the content or metadata being retrieved via MCP?
Stack Overflow does not have visibility into any content, queries, or metadata retrieved through the MCP Server. All retrieval occurs within the customer's Stack Overflow Internal tenant under existing permissions and controls. Stack Overflow may access limited information only as permitted under the customer's contract - for example, to provide technical support or aggregated, anonymized analytics - and never for model training or unrelated purposes.

How is access scoped (e.g., user-based, team-based, org-wide)?
Access is user-based and team-scoped, following the same permissions already defined in your Stack Overflow Internal site. The MCP Server uses OAuth tokens that inherit each user's roles and visibility settings, ensuring agents can only read or write to content that user is authorized to access.

Can we restrict which Stack Overflow Internal or collections of knowledge are accessible through MCP?
Yes. Access always mirrors your existing Stack Overflow Internal permissions, and admins can enable/disable MCP at the org level in the Admin Dashboard. MVP supports team-based permission scoping so agents only see content the signed-in user can see; finer scoping (e.g., tag/collection limits) is under consideration for post-MVP.

Is data encrypted at rest and in transit?
Yes. All data is encrypted in transit using HTTPS/TLS, and any stored configuration or token data within your environment is encrypted at rest according to your own infrastructure's security policies. The MCP Server itself does not store or transmit unencrypted content or credentials.

How are tokens managed, rotated, or revoked?
OAuth tokens are issued, scoped, and managed through your Stack Overflow Internal site, not by Stack Overflow. Tokens can be revoked or rotated at any time via your existing Stack Overflow Internal or identity provider controls. The MCP Server simply passes and validates tokens—it does not store or manage them directly.

Authentication and authorization

How does OAuth 2.0 work in the MCP Server setup?
The MCP Server is an OAuth 2.1–compliant authorization server that supports **Dynamic Client Registration** and **PKCE**. It acts as an authorization server for AI clients and an OAuth client to the Stack Overflow Internal API v3. Clients register dynamically and use the Authorization Code flow to obtain access tokens scoped to the signed-in user's Stack Overflow Internal permissions, preventing unauthorized access or token reuse.

Does the MCP Server support SSO (SAML, OIDC) for enterprise authentication?
Yes. The MCP Server relies on the existing authentication configured for the organization's Stack Overflow Internal site. If the Stack Overflow Internal site uses SSO through SAML or OIDC, the MCP Server automatically honors that flow—users authenticate through the same identity provider, and all access tokens are issued under those SSO credentials.

How are MCP clients (AI tools, copilots, etc.) authenticated and registered?
The MCP Server supports OAuth 2.1 Dynamic Client Registration through the `/register` endpoint. Each enterprise tenant uses a shared `client_id`, while individual MCP clients register dynamically by providing their client name and redirect URIs. The server issues a registration access token that allows each client to manage its registration. Dynamic registration ensures controlled, auditable access and that only registered clients can initiate the Authorization Code + PKCE flow.

Security controls

What audit logging or monitoring is included?
The MCP Server logs key usage events for auditability and security oversight. Logged data includes request timestamps, client identity, tool and prompt usage, and error rates. These logs are used to support debugging, compliance, and operational monitoring. Additional analytics and tenant-level reporting capabilities are planned for future releases.

Can we review access logs to see which tools or users queried which content?
Yes. Basic usage and access logs—including which agents, users, and tools made MCP requests—are available through the Amplitude Dashboard. These logs help admins monitor activity and troubleshoot integrations. More granular content-level auditing is planned for post-MVP releases.

How does the system prevent unauthorized write access or data exfiltration?
By design, it's defense-in-depth:

- Runs in your environment
- OAuth 2.0 + PKCE, strict redirect-URI validation, CSRF state checks, and per-client registration prevent token abuse.
- User-scoped permissions: tokens map to Stack Overflow Internal roles; team-based scoping ensures agents can only act on content that user can access.
- Write controls: MVP limits writes to user-initiated flows; admins can enable/disable MCP and govern write access.
- Audit logging in Datadog for visibility into agent/tool activity and errors.

How is CSRF or token replay prevented in the OAuth flow?
The MCP Server uses PKCE (Proof Key for Code Exchange) with SHA-256 and cryptographically random code verifiers to prevent interception or replay of authorization codes. It also enforces strict redirect URI validation and CSRF protection via state parameters, ensuring tokens can only be used by the original requesting client.

What happens if an MCP client's credentials are compromised?
Risk is minimal. MCP clients use **short-lived OAuth tokens** and **PKCE**, so no long-term client secrets are exposed. If a registration access token or client registration is compromised, it can be revoked via the `/registration/{client_id}` endpoint. Because access tokens are tied to individual user permissions and expire automatically, compromised credentials have limited impact and cannot be reused to access other Stack Overflow Internal data.

Compliance and governance

Does this architecture comply with common security standards (e.g., SOC 2, GDPR)?
Yes. The MCP Server adheres to the same security, privacy, and compliance framework as Stack Overflow Internal, which is **SOC 2 Type II certified** and **GDPR compliant**. Because the MCP Server operates within each customer's Stack Overflow Internal tenant, all content access and processing remain covered under the same audited controls and certifications.

Is Stack Overflow signing any new data processing agreements for MCP usage?
No new data processing agreement is required. The MCP Server operates within each customer's existing Stack Overflow Internal tenant and does not transfer or store customer data outside that environment. All usage remains governed by the customer's existing Stack Internal Enterprise agreement and DPA.

How are internal compliance or data retention policies respected?
The MCP Server operates within the customer's dedicated Stack Overflow Internal tenant, where all content access and storage follow the same compliance and retention controls already in place for that tenant. The MCP Server does not replicate or persist content outside the Stack Overflow Internal environment, so existing organizational policies continue to apply.

How can our InfoSec team review or approve the deployment?
Your InfoSec team can review our architecture documentation before deployment. The server's OAuth 2.0–compliant design and no-data-exfiltration approach make it straightforward to validate against internal security and compliance standards.

Legal and contractual

Is there a separate agreement or amendment required for MCP Server use?
No. The MCP Server is covered under your existing Stack Internal Enterprise agreement. During the free trial, standard Stack Overflow Internal terms apply; a commercial order form is only required if you choose to continue access through the Premium API subscription after the trial ends.

Is the free trial covered under our existing Enterprise terms?
Not automatically. The MCP Server is part of the Premium API offering, which is an add-on to your existing Stack Internal Enterprise contract. To enable MCP Server, your account team will provide an updated order form reflecting the Premium API subscription, but no new legal agreement or DPA is required.

What happens to any data or logs collected during the pilot or trial period?
All data and logs generated during the pilot or trial remain within your environment. Stack Overflow does not store, access, or retain any customer data from MCP activity. Any telemetry collected (e.g., usage metrics in Datadog) is for system performance monitoring and can be deleted or disabled at your discretion.

How is intellectual property handled if AI agents generate or update content?
Any content created or updated through MCP—whether by a developer or AI agent—belongs to your organization under your existing Stack Internal Enterprise terms. Attribution is preserved automatically, so each contribution is linked to the authenticated user or agent that submitted it.

Is the MCP Server open source, and can it be extended?
Yes. The MCP Server implementation is open source, allowing customers and partners to review its code and understand how it works. While Stack Overflow hosts and manages the production version used with Stack Overflow Internal, the open-source project can be referenced or extended for internal development or integrations.

Technical architecture and deployment

Integration

How does MCP connect to our Stack Overflow Internal site?
The MCP Server connects to your Stack Overflow Internal site through the Stack Overflow Internal API v3 using OAuth 2.0 authentication. It serves as a secure bridge between your Stack Overflow Internal data and approved AI tools, enabling read and write access according to the permissions defined in your Stack Overflow Internal environment.

Which APIs or endpoints does it use?
The MCP Server uses the Stack Overflow Internal API v3 and exposes structured MCP endpoints such as `/search`, `/get_content`, `/post_article`, and `/record_feedback`. These map directly to Stack Overflow Internal resources like questions, answers, and articles, ensuring agents access content securely and consistently through standardized APIs.

Does it support read-only and read/write flows?
Yes. The MCP Server supports both. The MVP release includes read and user-initiated write flows—such as drafting new questions or articles—while agent-initiated updates and additional write types (e.g., edits, comments, votes) are planned for post-MVP releases.

How does it handle versioning when Stack Overflow updates its API?
The MCP Server is built on the Stack Overflow Internal API v3 and designed for compatibility with future versions. Because it's open source, updates to the API or MCP specification can be adopted quickly—either by Stack Overflow or by your own team—without disrupting existing integrations

Supported agents/tools

Which AI agents currently support MCP (e.g., Cursor, Windsurf, JetBrains AI Assistant)?
MCP is already supported by several AI developer tools, including Github Copilot, Cursor, Windsurf, JetBrains AI Assistant and Junie, Chatgpt, Anthropic, and Microsoft. Broader ecosystem adoption is growing, MCP is quickly becoming the standard

Is GitHub Copilot supported today?
Yes, Github Copiilot can connect MCP servers via the IDE (commonly VS Code)

How does MCP handle future integration with proprietary copilots or chatbots?
MCP uses an open, standardized protocol, making it straightforward to integrate with proprietary copilots or internal chatbots. Your teams can register these tools as MCP clients via OAuth, giving them secure, scoped access to Stack Overflow Internal data—no custom API work required. This flexibility ensures future compatibility as new AI tools adopt MCP.

Deployment

Can we deploy the MCP Server locally, in our own cloud, or via Cloudflare?
Not currently. The MCP Server is hosted and managed by Stack Overflow as part of each customer's Stack Overflow Internal tenant. It is not available for self-hosting or third-party deployment. Customers interact with their dedicated endpoint (for example, https://yourtenant.stackenterprise.co/mcp) to ensure consistent security, compliance, and support.

What are the infrastructure requirements?
None. The MCP Server is fully hosted and managed by Stack Overflow as part of each customer's Stack Overflow Internal tenant. Customers connect to their dedicated MCP endpoint (for example, https://yourtenant.stackenterprise.co/mcp)—no separate deployment, database, or infrastructure is required. All data access and permissions follow existing Stack Overflow Internal configuration and security controls.

How long does setup typically take?
Setup is quick - only a few clicks. Because the MCP Server is already hosted within each customer's Stack Overflow Internal tenant, setup only involves connecting approved AI tools, completing OAuth authorization, and validating access. No separate deployment or infrastructure configuration is required.

Are there prebuilt configuration templates or scripts?
Yes. Stack overflow provides one click add-ons to your favorite AI tools like Cursor and VS Code.

Monitoring and analytics

What telemetry is collected through MCP?
MCP collects lightweight, non-content telemetry for monitoring and performance—such as total requests, active agents, errors, and tool usage. No message content or query data is stored.

How can admins monitor agent usage, API calls, or contribution patterns?
Today admins cannot self service usage analytics. This will be addressed in the roadmap.

Performance and reliability

What are the latency expectations between AI tools and Stack Overflow Internal data?
Latency is typically sub-second to a few seconds, depending on network distance and API query complexity. Because the MCP Server runs within your environment and communicates directly with your Stack Overflow Internal API, response times are optimized and avoid any external routing through Stack Overflow infrastructure.

Are there rate limits or throttling mechanisms in place?
Yes. The MCP Server inherits the rate limits of your Stack Overflow Internal API v3 and can apply additional per-agent or per-user throttling if configured. These safeguards prevent excessive query volume and ensure consistent performance across connected tools.

How does the MCP Server handle downtime or API failures?
The MCP Server includes graceful error handling and retry logic for transient API issues. If your Stack Overflow Internal API is unavailable, the server returns standardized error responses to connected agents without exposing sensitive data. Because it's hosted in your environment, uptime and recovery align with your existing infrastructure and monitoring practices.

Product and use case

Functionality

What's the difference between the MCP Server and the Premium API?
The Premium API provides direct, high-volume programmatic access to Stack Overflow Internal data, while the MCP Server adds an AI-ready integration layer on top of that data. MCP exposes structured tools and prompts that let AI agents securely read, write, and interact with Stack Overflow Internal content—enabling copilots, IDE assistants, and internal bots to use your knowledge base in real time.

What capabilities are included in the initial release vs. future roadmap (e.g., agent-initiated writes)?
The initial release includes:

- Secure OAuth 2.0 authentication
- Read access to Stack Overflow Internal content (questions, answers, articles)
- User-initiated write flows (drafting questions or articles)
- Usage logging and analytics via Datadog
- Admin controls for enabling/disabling MCP

The future roadmap adds:

- Agent-initiated writes and updates
- Additional tooling / user prompts
- Granular admin dashboards for permissions and analytics

Can agents propose updates to content or create new questions/articles?
Yes. In the GA release, agents can propose new questions or articles through user-initiated flows—developers approve and publish the drafts.

How does attribution work when AI agents surface or generate content?
The MCP Server includes attribution metadata with every response from Stack Overflow Internal. When AI agents surface or generate content, they are prompted to display the original question, answer, or article link from the Stack Overflow Internal source. This ensures transparency, preserves author credit, and maintains traceability back to verified knowledge.

Administration

What can admins control in the MCP Dashboard?
Admins can control if their organization is able to use the MCP server as an "On/Off" toggle is present. Additional functionality like per client restrictions will be added

Can we disable or limit access per agent or per user?
Not at this time. Access per agent or user is a future development.

Are there role-based permissions or scopes?
Yes. The MCP Server enforces role-based permissions and scoped OAuth access aligned with your Stack Overflow Internal site. Permissions are inherited from each user's Stack Overflow Internal role, ensuring agents can only read or write to content their associated users are authorized to access.

Additional questions? Reach out to your Customer Success Manager or Stack Overflow product support.

If you need further support or have questions, contact your site administrator.