GitHub Integration

Roy OS Integration Guide | READ-ONLY

Overview

The Roy OS GitHub integration allows the meeting agent to read source code and documentation from specific repositories during meetings. This powers codebase Q&A — your team can ask the agent about architecture, recent changes, or implementation details during technical discussions. The integration is strictly read-only.

What Roy Can Do in GitHub

Capability	Description
Read file contents	Reads source code files, READMEs, and documentation from connected repositories.
Browse directory trees	Navigates repository file structure to find relevant code for meeting discussions.
Read PR descriptions	Reads pull request descriptions and metadata to provide context during code review meetings.

> **What Roy cannot do:** Push code, create branches, open PRs, write comments, modify issues, access secrets, trigger Actions workflows, manage webhooks, or perform any write or admin operations. The integration is read-only by design.

Required Permissions

GitHub App Permission	Why It's Needed
`contents:read`	Read file contents, directory trees, and README files
`metadata:read`	Read repository name, description, and default branch (required for all GitHub Apps)

Setup Guide

Prerequisites

You'll need GitHub organization admin access and access to your Azure Key Vault.

1 **Create a GitHub App**

In your GitHub organization → **Settings** → **Developer settings** → **GitHub Apps** → **New GitHub App**. Name it "Roy OS". Set the homepage URL and webhook URL provided by Roy AI during onboarding.

2 **Configure Permissions**

Under **Permissions**, set **Repository permissions**: `Contents` → **Read-only**, `Metadata` → **Read-only**. Leave all other permissions at "No access."

3 **Generate a Private Key**

Under **General** → **Private keys** → **Generate a private key**. A `.pem` file downloads. Also note the **App ID** from the General section.

4 **Store Credentials in Key Vault**

Secret Name	Value
`github-app-id`	Numeric App ID
`github-private-key`	Contents of the `.pem` private key file

5 **Install on Specific Repositories**

Go to the GitHub App's **Install App** page → select your organization → choose **"Only select repositories"** and pick the repos your teams need during meetings. You can add or remove repos at any time.

6 **Verify Connectivity**

Roy AI generates an installation token from the private key and makes a test API call to read from one of the connected repos.

Managing Access

Action	How
Add a repository	GitHub org admin → App installation settings → add the repo. Immediate effect.
Remove a repository	GitHub org admin → App installation settings → deselect the repo. Immediate effect.

Revoke all access	Uninstall the GitHub App from your organization. All installation tokens invalidated.
Rotate credentials	Generate a new private key in GitHub App settings (old key is immediately invalidated). Update Key Vault.

FAQ

Can Roy access all repos in my organization?

No. During installation, you explicitly select which repositories the app can access. Roy cannot see repos outside the installation scope. You control the list.

Can Roy push code or modify anything?

No. The app only has `contents:read` and `metadata:read` permissions. GitHub enforces this at the API level — write requests would be rejected even if attempted.

How does authentication work?

Roy OS uses the GitHub App private key to generate short-lived installation tokens (valid for 1 hour). No long-lived tokens are stored. A new token is generated as needed.

> **Audit trail:** Every GitHub API call is logged (repo, file path, action, timestamp) in Azure Log Analytics. GitHub also logs app activity in your organization's audit log.