Source Code Management Platform Configuration Best Practices

by the Open Source Security Foundation (OpenSSF) Best Practices Working Group, 2023-08-29

1. Intro

Collaborative source code management platforms (such as GitHub and GitLab) play a critical role in modern software development, providing a central repository for storing, managing, and versioning source code as well as collaborating with a community of developers. However, they also represent a potential security risk if not properly configured. In this guide, we will explore the best practices for securing these platforms, covering topics that include user authentication, access control, permissions, monitoring, and logging.

2. Audience

This guide has been written for the:

- **Maintainer** who wants to improve the security posture for one or more GitHub repositories or GitLab projects they support.
- **Owner** who wants to improve the security posture for their GitHub organization or GitLab group they manage.
- **Open Source Program Office (OSPO)** (or a team that plays a similar role) who is typically responsible for multiple GitHub organizations or GitLab groups.
- **Operations** team tasked with applying policies as part of their work managing assets on these platforms.
- **GitHub/GitLab enterprise administrator** who wants to improve the security posture for their SCM enterprise.

3. Tooling

Below is a non-exhaustive list of possible tools that can be used to assist in review source code repositories.

3.1. Allstar - https://github.com/ossf/allstar

An open-source project from the OpenSSF that scans GitHub organizations for “repository level” misconfigurations. Allstar detects a subset of the “repository level” policies suggested by this document. It can be configured to scan all repositories in an organization or a subset of them and is supported by the following SCMs:

- GitHub Cloud

3.2. Legitify - https://github.com/Legit-Labs/legitify

An open-source project from Legit Security that scans SCM assets to find misconfigurations, security issues, and unfollowed best practices. Legitify detects all policies suggested by this document and supports the following SCMs:

- GitHub Cloud
- GitHub Enterprise Server
- GitLab Cloud
- GitLab Server

3.3. Scorecard - https://github.com/ossf/scorecard

An open-source project from the OpenSSF that scans repositories for security issues and provides security health metrics. Scorecard detects many of the “repository level” policies suggested by this document and supports the following SCMs:

- GitHub Cloud
- GitHub Enterprise Server
- GitLab Cloud
- GitLab Server

4. Recommendations

Each specific recommendation below is noted to be applicable to either GitHub or GitLab by use of an appropriate icon and text, and is linked to the detailed best practice definition if available:

- (Applies to: GitHub GitLab)
    - or
- GitHub GitLab

For recommendations only applicable to GitHub or GitLab visit one of the following pages:

- GitHub Recommendations
- GitLab Recommendations

4.1. Continuous Integration / Continuous Deployment

- Workflows Should Not Be Allowed To Approve Pull Requests GitHub
- GitHub Actions Should Be Restricted To Selected Repositories GitHub
- GitHub Actions Should Be Limited To Verified or Explicitly Trusted Actions GitHub

- Default Workflow Token Permission Should Be Read Only GitHub
- Runner Group Should Be Limited to Private Repositories GitHub
- Runner Group Should Be Limited to Selected Repositories GitHub

4.2. Enterprise

- Two-Factor Authentication Should Be Enforced For The Enterprise GitHub
- Enterprise Should Not Allow Members To Create public Repositories GitHub
- Enterprise Should Not Allow Members To Invite Outside Collaborators GitHub
- Enterprise Should Not Allow Members To Change Repository Visibility GitHub
- Enterprise Should Use Single-Sign-On GitHub
- Enterprise Should Not Allow Members To Fork Internal And Private Repositories GitHub
- Two-Factor Authentication Should Be Enforced For The Group GitLab
- Forking of Repositories to External Namespaces Should Be Disabled. GitLab
- Group Should Enforce Branch Protection GitLab
- Webhooks Should Be Configured To Use SSL GitLab

4.3. Members, Access Control and Permissions

- Organization Should Have Fewer Than Three Owners GitHub
- Organization Admins Should Have Activity In The Last 6 Months GitHub
- Organization Members Should Have Activity In The Last 6 Months GitHub
- Two Factor Authentication Should Be Enabled for Collaborators GitLab
- Two Factor Authentication Should Be Enabled for External Collaborators GitLab
- Administrators Should Have Activity in the Last 6 Months GitLab

4.4. Repository

- Repository Should Be Updated At Least Quarterly GitHub
- Workflows Should Not Be Allowed To Approve Pull Requests GitHub
- Default Branch Should Require Code Review GitHub GitLab
- Default Workflow Token Permission Should Be Set To Read Only GitHub
- Default Branch Should Be Protected GitHub GitLab
- Default Branch Should Not Allow Force Pushes GitHub GitLab
- Default Branch Should Require Code Review By At Least Two Reviewers GitHub GitLab
- Vulnerability Alerts Should Be Enabled GitHub
- OpenSSF Scorecard Score Should Be Above 7 GitHub

- GitHub Advanced Security – Dependency Review Should Be Enabled For A Repository GitHub
- Default Branch Deletion Protection Should Be Enabled GitHub
- Default Branch Should Require Linear History GitHub
- Default Branch Should Require All Checks To Pass Before Merge GitHub
- Default Branch Should Require Branches To Be Up To Date Before Merge GitHub
- Repository Should Have Fewer Than Three Admins GitHub
- Default Branch Should Restrict Who Can Push To It GitHub
- Default Branch Should Require All Commits To Be Signed GitHub GitLab
- Webhooks Should Be Configured With A Secret GitHub
- Webhooks Should Be Configured To Use SSL GitHub
- Default Branch Should Require All Conversations To Be Resolved Before Merge GitHub
- Default Branch Should Restrict Who Can Dismiss Reviews GitHub
- Default Branch Should Require New Code Changes After Approval To Be Re-Approved GitHub GitLab
- Default Branch Should Limit Code Review to Code-Owners GitHub GitLab
- Forking Should Not Be Allowed for This Repository GitHub
- Project Should Be Updated At Least Quarterly GitLab
- Repository Should Not Allow Review Requester To Approve Their Own Request GitLab
- Merge Request Authors Should Not Be Able To Override the Approvers List GitLab
- Project Should Require All Pipelines to Succeed GitLab
- Forking Should Not Be Allowed GitLab
- Project Should Require All Conversations To Be Resolved Before Merge GitLab
- Repository Should Not Allow Committer Approvals GitLab
- Webhook Configured Without SSL Verification GitLab
- Project Should Have Fewer Than Three Owners GitLab

4.5. Operations

General Recommendations

- Organization Management Should Be Consolidated Under a Central Account. (Applies to: GitHub)
- Organization Membership Should Be Limited to Its Staff When Relevant. (Applies to: GitHub)
- Review Security Policies and Procedures At Least Annually. (Applies to: GitHub GitLab)
- Establish a Clear Communication and Incident Response Plan. (Applies to: GitHub GitLab)
- Conduct Regular Security Audits and Vulnerability Assessments. (Applies to: GitHub GitLab)

- Use Insights to Track Activity and in Repositories and Organizations. (Applies to: GitHub)
- Use Tools Built On APIs to Automate Tasks and Avoid Needing Elevated Privileges. (Applies to: GitHub GitLab)
- Review the Configuration Settings Before Making a Repository Public. (Applies to: GitHub GitLab)
- Review the Configuration Settings After Transferring a Repository into the Organization. (Applies to: GitHub GitLab)
- Provide Automated Alerts and Tooling to Ensure Ongoing Compliance. (Applies to: GitHub GitLab)
- Review Audit Logs to Track Activity and Changes in Repositories and Organizations. (Applies to: GitHub)
- Group Membership Should Be Limited to Organization Staff When Relevant. (Applies to: GitHub)
- Review Audit Events to Track Activity and Changes in Projects and Groups. (Applies to: GitHub)

Specific Recommendations

- Two-Factor Authentication Should Be Enforced For The Organization GitHub
- Organization Should Use Single-Sign-On GitHub
- Default Member Permissions Should Be Restricted GitHub
- Only Admins Should Be Able To Create Public Repositories GitHub
- Webhooks Should Be Configured To Use SSL GitHub GitLab
- Webhooks Should Be Configured With A Secret GitHub
- Configure Security Alerts and Vulnerability Scanning at the Organization or Repository Level. (Applies to: GitHub)
- Enable GitHub Advanced Security features for Private and Internal Repositories. (Applies to: GitHub)
- Two-Factor Authentication Should Be Enforced For The Group GitLab
- Group Should Use Single-Sign-On (Applies to: GitLab)
- Only Admins Should Be Able To Create Public Projects and Groups. (Applies to: GitLab)

5. Acknowledgements

The following community members helped contribute to this guidance:

- Avishay Balter, Microsoft
- Chris de Almeida, IBM
- Christine Abernathy, F5 - project lead
- Daniel Appelquist, Snyk - project lead

- Noam Dotan, Legit Security - project lead
- David A. Wheeler, The Linux Foundation