Configure Gitleaks

BRAIN STATION 23

Version: 1.0

October 2024

(**Privacy:** Private)

Table of Contents

1 Introduction

Gitleaks can be integrated with Git, GitHub, GitLab, and Bitbucket to perform automated scanning for sensitive data across these platforms. Following sections describing the configurations.

2 Using Gitleaks Locally with Git

For local Git repositories, you can set up Gitleaks to scan before committing code. This is useful if you want developers to run checks before pushing code to remote repositories.

- **Install Gitleaks**: Download and install it locally on your development machines.
- **Pre-commit Hook**: Add Gitleaks as a Git pre-commit hook to automatically scan code each time a commit is made.

```
# .git/hooks/pre-commit
gitleaks detect --source . --exit-code 1
```

This setup will block commits if Gitleaks finds any sensitive data.

3 Integrating Gitleaks with GitHub Actions

GitHub Actions allows you to integrate Gitleaks as part of your CI pipeline. This setup enables Gitleaks to scan each pull request or push event.

- **GitHub Workflow Configuration**: Add a workflow file (e.g., `.github/workflows/gitleaks.yml`) to your GitHub repository:

```
name: Gitleaks Scan

on:
  push:
    branches:
      - main
  pull_request:

jobs:
  gitleaks:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout code
        uses: actions/checkout@v2
      - name: Run Gitleaks
        uses: zricethezav/gitleaks-action@v2
        with:
          args: detect --source . --exit-code 1
```

- **Alerts**: Configure alerts for any findings to notify your team in GitHub, email, or your preferred communication channel.

4 Integrating Gitleaks with GitLab CI/CD

In GitLab, you can set up Gitleaks in the CI/CD pipeline to scan each commit or merge request.

- **GitLab CI Configuration**: Add Gitleaks as a job in `.gitlab-ci.yml`:

```
gitleaks-scan:
  image: zricethezav/gitleaks
  script:
    - gitleaks detect --source . --exit-code 1
  allow_failure: false
```

- **Alerts and Visibility**: If Gitleaks finds sensitive data, the pipeline will fail, and the results will be visible in the GitLab pipeline logs.

5 Integrating Gitleaks with Bitbucket Pipelines

For Bitbucket, Gitleaks can be configured in the `bitbucket-pipelines.yml` file to run on each push or pull request.

- **Bitbucket Pipeline Configuration**: Add Gitleaks to the pipeline file:

```
image: atlassian/default-image:2

pipelines:
  default:
    - step:
        name: Run Gitleaks
        script:
          - curl -sSfL
https://github.com/zricethezav/gitleaks/releases/latest/download/gitleak
s-linux-amd64 -o gitleaks
          - chmod +x gitleaks
          - ./gitleaks detect --source . --exit-code 1
```

- **Alerts and Blocking**: If sensitive data is detected, the pipeline will fail, alerting the team to review the changes.

6 Centralized Scanning for All Repositories

To monitor all repositories (especially in GitHub or GitLab organizations or Bitbucket workspaces):

- **Use Gitleaks in CI Pipelines Globally**: Add Gitleaks to the pipelines of all projects to catch sensitive data across the organization.
- **Run Gitleaks Periodically**: Schedule periodic scans of all repositories to catch any sensitive information that might have been missed in prior scans.