gitsec Documentation

Roberto Abdelkader Martínez Pérez

Jun 28, 2018

Contents:

gitsec is an *automated secret discovery service* for **git** that helps you detect sensitive data leaks.

gitsec doesn't directly detect sensitive data but uses already available open source tools with this purpose and provides

a framework to run them as one.

CHAPTER 1

Architecture

gitsec is build upon buildbot and buildbot-washer therefore inheriting their architecture.

Master processes receive code changes from git repositories. When a change is detected, workers are spawned to run the defined plugins on the configuration file(s).

The **master** process runs on a *docker* container and spawns **workers** in new containers as needed. The master process is a regular *buildbot* master with gitsec's specific configuration. Worker processes are *buildbot* worker processes with an specific *buildbot-washer* task registered.

CHAPTER 2

Plugins

Project	Image	Summary
api-key-detect	bbvalabsci/gitsec-api-key-detect	Scan a codebase for API keys and passwords
git-hound	bbvalabsci/gitsec-git-hound	Git plugin that prevents sensitive data from being committed
git-secrets	bbvalabsci/gitsec-git-secrets	Prevents you from committing secrets and credentials into git repositories
gittyleaks	bbvalabsci/gitsec-gittyleaks	Find sensitive information for a git repo
trufflehog	bbvalabsci/gitsec-trufflehog	Searches through git repositories for high entropy strings and secrets, digging deep into commit history

CHAPTER 3

Usage

In order to use **gitsec** you must follow these steps:

1. Configure and deploy a master.
2. Configure your GitHub repository or organization webhooks.
3. (Optional) Add a *.gitsec.yml* configuration file to your project.

3.1 Master Deployment

You can run the gitsec master process with docker this way:

```
docker run -ti -v/var/run/docker.sock:/var/run/docker.sock -p8010:8010 -p9989:9989␣
↪bbvalabsci/gitsec
```

Master configuration documentation is available here.

3.2 Github Webhook Integration

You can set a Github webhook to trigger a *gitsec* analysis for a particular repository or for all the repositories in an organization.

Follow this guide to add the webhook.

You should set a strong *secret* to the webhook, remember to pass the secret to your master using the **GITHUB_HOOK_SECRET** variable.

The endpoint to point to shall be "http://YOUR-HOST-AND-POR-HERE/change_hook/github".

CHAPTER 4

Configuration File Format

gitsec configuration is at least one YAML file defining the list of plugins to run for each source code change.

Two configuration files may be defined: one in the server, another in the user's repository. The former, if present, is managed by the owner of the gitsec service and contains the list of plugins that must always run for a code change. The latter is managed by the source code repository owners and contains an extra list of plugins and configuration for that specific repository.

This way a list of plugins may be enforced by the gitsec service owner and, at the same time, maintains flexibility for the developers to add their own checks.

The configuration file format is YAML.

This is an example of configuration file:

```
plugins:
    bbvalabsci/gitsec-git-secrets:
        options:
            prohibited:
                password:
                    value: '^password:'
                    type: regex
    bbvalabsci/gitsec-api-key-detect:
        unimportant: yes
    bbvalabsci/gitsec-trufflehog:
    bbvalabsci/gitsec-gittyleaks:
```

- The *plugins* key contains the list of plugins. In the example 4 plugins are defined.
  - Each plugin section is defined by the name of the **docker image to run**.
    - The plugin section may contain the following keys:
      - *unimportant* (yes|no): If **yes** the failure of this plugin will not make the whole check to fail.
      - *options*: The parameter passed to the plugin. Depends on the plugin.

CHAPTER 5

Create a New Plugin

If you want to create a gitsec plugin for a tool of yours, or for any other already available tool, follow the instructions given here.

If you want your plugin to be part of gitsec distribution, please send a **pull request** adding the plugin files in a directory under the *plugins* directory.

CHAPTER 6

I’ve just committed a secret! How I fix it??

https://help.github.com/articles/removing-sensitive-data-from-a-repository/

CHAPTER 7

Indices and tables

- genindex
- modindex
- search