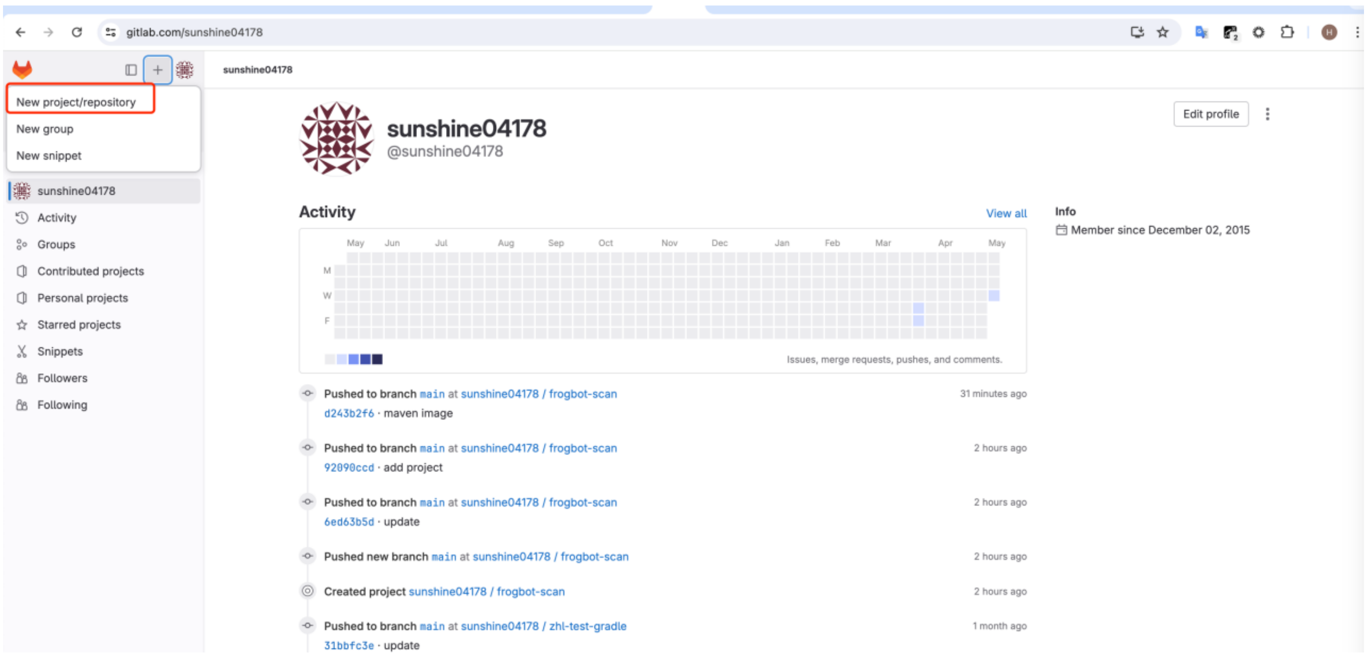
1.Create gitlab project


gitlab.com/sunshine04178

sunshine04178

New project/repository
New group
New snippet

sunshine04178
Activity
Groups
Contributed projects
Personal projects
Starred projects
Snippets
Followers
Following

sunshine04178
@sunshine04178

Edit profile

Activity — View all

Info
Member since December 02, 2015

Issues, merge requests, pushes, and comments.

Pushed to branch main at sunshine04178 / frogbot-scan — 31 minutes ago
d243b2f6 · maven image

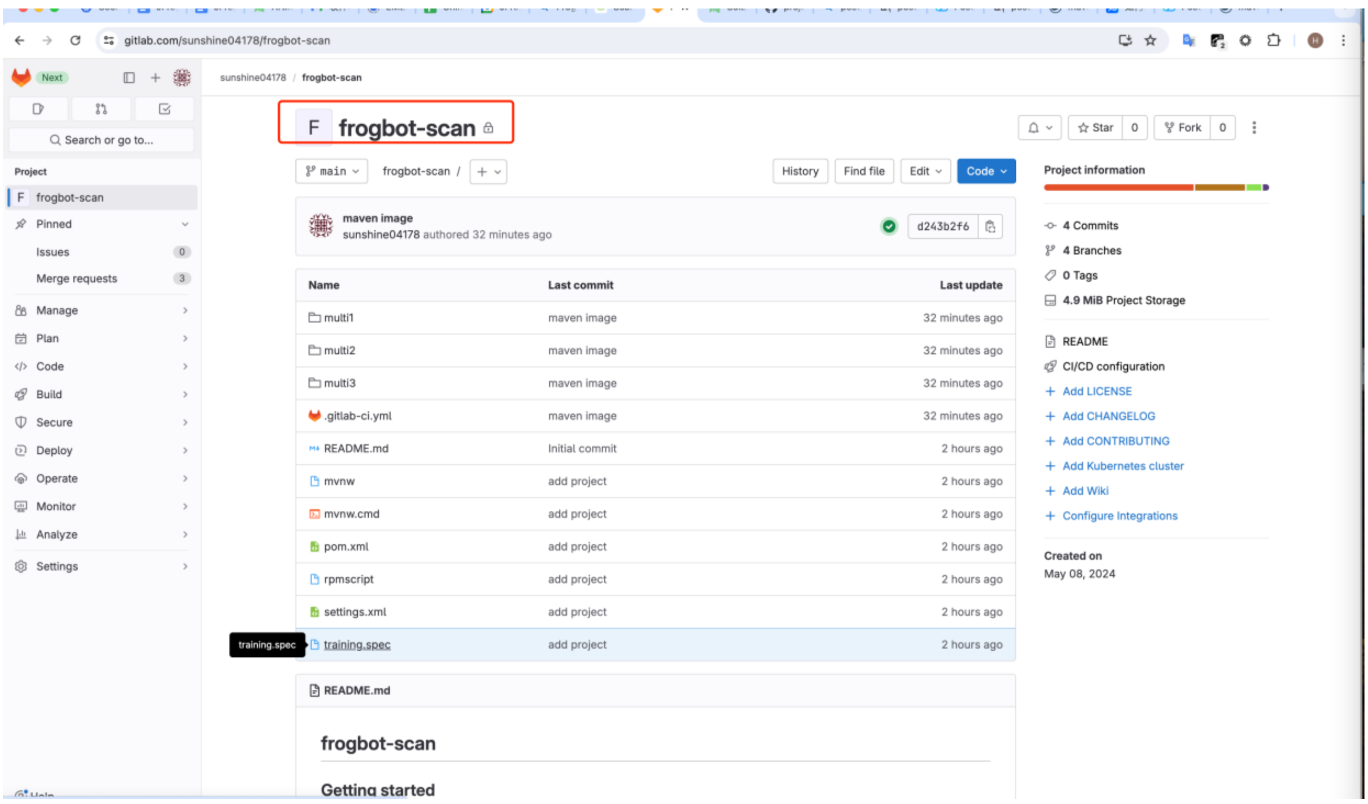
Pushed to branch main at sunshine04178 / frogbot-scan — 2 hours ago
92090ccd · add project

Pushed to branch main at sunshine04178 / frogbot-scan — 2 hours ago
6ed63b5d · update

Pushed new branch main at sunshine04178 / frogbot-scan — 2 hours ago

Created project sunshine04178 / frogbot-scan — 2 hours ago

Pushed to branch main at sunshine04178 / zhl-test-gradle — 1 month ago
31bbfc3e · update


gitlab.com/sunshine04178/frogbot-scan

sunshine04178 / frogbot-scan

Project: frogbot-scan, Pinned, Issues 0, Merge requests 3, Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, Analyze, Settings

F frogbot-scan

main / frogbot-scan / — History · Find file · Edit · Code

maven image
sunshine04178 authored 32 minutes ago — d243b2f6

Name	Last commit	Last update
multi1	maven image	32 minutes ago
multi2	maven image	32 minutes ago
multi3	maven image	32 minutes ago
.gitlab-ci.yml	maven image	32 minutes ago
README.md	Initial commit	2 hours ago
mvnw	add project	2 hours ago
mvnw.cmd	add project	2 hours ago
pom.xml	add project	2 hours ago
rpmscript	add project	2 hours ago
settings.xml	add project	2 hours ago
training.spec	add project	2 hours ago

Project information
4 Commits
4 Branches
0 Tags
4.9 MiB Project Storage
README
CI/CD configuration
Add LICENSE
Add CHANGELOG
Add CONTRIBUTING
Add Kubernetes cluster
Add Wiki
Configure Integrations

Created on
May 08, 2024

README.md

frogbot-scan

Getting started

2.Download maven project and copy to your project of frogbot-scan

```
git clone https://gitlab.com/sunshine04178/frogbot-scan.git  (your project
address)
unzip project-examples-master.zip
mv project-examples-master/maven-examples/maven-example/*
/Users/hailangz/test/frogbot-scan
```

3.Add yml

```
cd /Users/hailangz/test/frogbot-scan

vim .gitlab-ci.yml
```

```
frogbot-scan:
image: maven:3.6.3
rules:
- if: $CI_PIPELINE_SOURCE == 'merge_request_event'
when: manual
variables:
FROGBOT_CMD: "scan-pull-request"
JF_GIT_BASE_BRANCH: $CI_MERGE_REQUEST_TARGET_BRANCH_NAME
# Repository scanning is triggered by any push to the default branch.
# If you'd like a different branch to be scanned, replace
$CI_DEFAULT_BRANCH in the line below with the name of the branch, wrapped
with quotes (").
- if: $CI_COMMIT_BRANCH == $CI_DEFAULT_BRANCH || $CI_PIPELINE_SOURCE ==
"schedule"
variables:
FROGBOT_CMD: "scan-repository"
JF_GIT_BASE_BRANCH: $CI_COMMIT_BRANCH
variables:
# [Mandatory]
# JFrog platform URL (This functionality requires version 3.29.0 or above
of Xray)
JF_URL: $JF_URL
# [Mandatory if JF_USER and JF_PASSWORD are not provided]
# JFrog access token with 'read' permissions for Xray
JF_ACCESS_TOKEN: $JF_ACCESS_TOKEN
# [Mandatory if JF_ACCESS_TOKEN is not provided]
# JFrog user and password with 'read' permissions for Xray
# JF_USER: $JF_USER
# JF_PASSWORD: $JF_PASSWORD
# [Mandatory]
# GitLab access token. Ensure the token has the following permissions,
depedending on your GiLab deployment type:
# Self hosted: api, read_api, read_user, read_repository.
# Cloud: api, read_api, read_repository
JF_GIT_TOKEN: $USER_TOKEN
# Predefined GitLab variables. There's no need to set them.
JF_GIT_PROVIDER: gitlab
JF_GIT_OWNER: $CI_PROJECT_NAMESPACE
JF_GIT_REPO: $CI_PROJECT_NAME
JF_GIT_PULL_REQUEST_ID: $CI_MERGE_REQUEST_IID
# [Optional, default: https://gitlab.com]
# API endpoint to GitLab
# JF_GIT_API_ENDPOINT: https://gitlab.example.com
# [Optional]
# By default, the Frogbot workflows download the Frogbot executable as
well as other tools
```

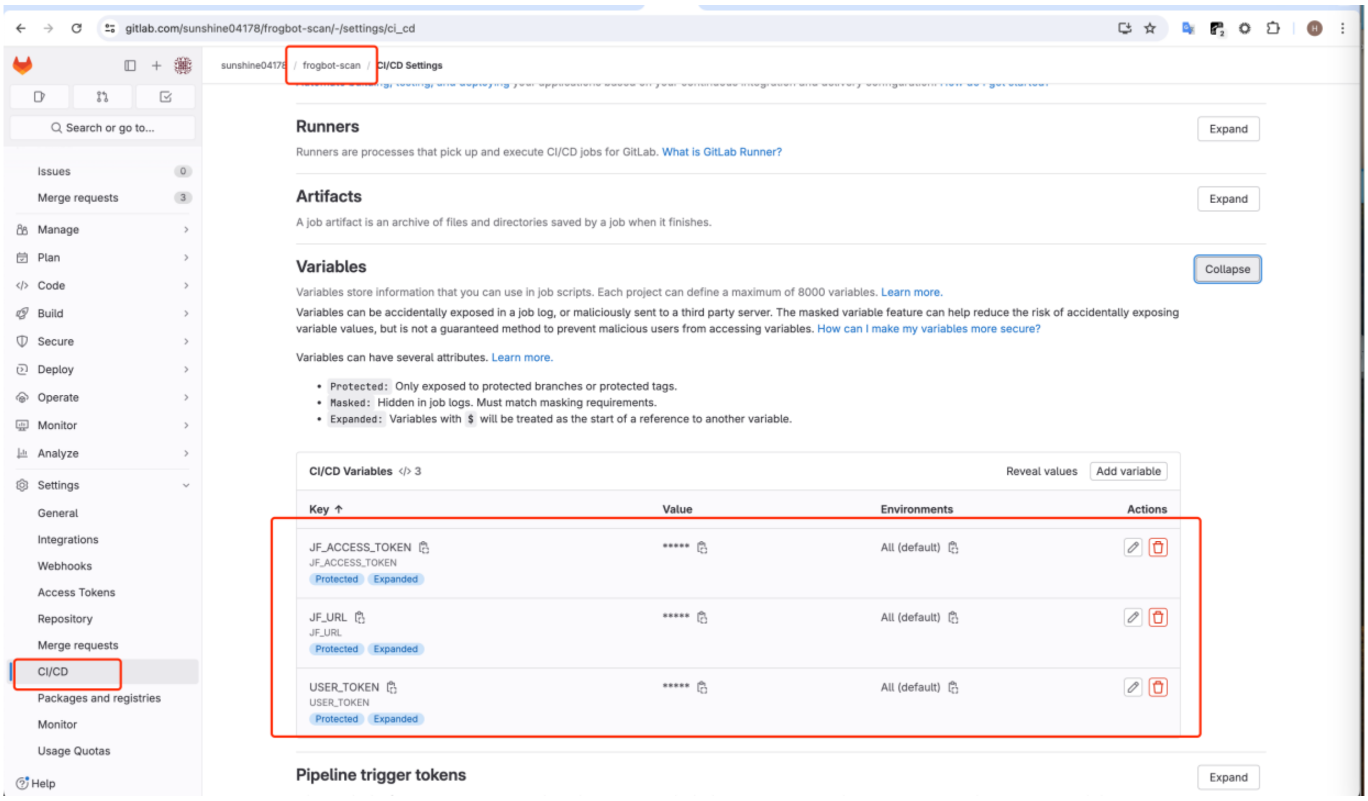
```
  # needed from https://releases.jfrog.io
  # If the machine that runs Frogbot has no access to the internet, follow
these steps to allow the
  # executable to be downloaded from an Artifactory instance, which the
machine has access to:
  #
  # 1. Login to the Artifactory UI, with a user who has admin credentials.
  # 2. Create a Remote Repository with the following properties set.
  # Under the 'Basic' tab:
  # Package Type: Generic
  # URL: https://releases.jfrog.io
  # Under the 'Advanced' tab:
  # Uncheck the 'Store Artifacts Locally' option
  # 3. Set the value of the 'JF_RELEASES_REPO' variable with the Repository
Key you created.
  # JF_RELEASES_REPO: ""
  # [Optional]
  # Configure the SMTP server to enable Frogbot to send emails with detected
secrets in pull request scans.
  # SMTP server URL including should the relevant port: (Example:
smtp.server.com:8080)
  # JF_SMTP_SERVER: ""
  # [Mandatory if JF_SMTP_SERVER is set]
  # The username required for authenticating with the SMTP server.
  # JF_SMTP_USER: ""
  # [Mandatory if JF_SMTP_SERVER is set]
  # The password associated with the username required for authentication
with the SMTP server.
  # JF_SMTP_PASSWORD: ""
  ##########################################################################
  #
  ## If your project uses a 'frogbot-config.yml' file, you should define ##
  ## the following variables inside the file, instead of here. ##
  ##########################################################################
  #
  # [Mandatory if the two conditions below are met]
  # 1. The project uses yarn 2, NuGet, or .NET to download its dependencies
  # 2. The `installCommand` variable isn't set in your frogbot-config.yml
file.
  #
  # The command that installs the project dependencies (e.g "nuget restore")
  JF_INSTALL_DEPS_CMD: ""
  # [Optional, default: "."]
  # Relative path to the root of the project in the Git repository
  # JF_WORKING_DIR: path/to/project/dir
  # [Default: "*.git*;*node_modules*;*target*;*venv*;*test*"]
  # List of exclusion patterns (utilizing wildcards) for excluding paths in
the source code of the Git repository during SCA scans.
  # JF_PATH_EXCLUSIONS: "*.git*;*node_modules*;*target*;*venv*;*test*"
  # [Optional]
  # Xray Watches. Learn more about them here:
https://www.jfrog.com/confluence/display/JFROG/Configuring+Xray+Watches
  # JF_WATCHES: <watch-1>,<watch-2>...<watch-n>
  # [Optional]
  # JFrog project. Learn more about it here:
  https://www.jfrog.com/confluence/display/JFROG/Projects
  # JF_PROJECT: <project-key>
  # [Optional, default: "FALSE"]
  # Displays all existing vulnerabilities, including the ones that were
  added by the pull request.
  # JF_INCLUDE_ALL_VULNERABILITIES: "TRUE"
  # [Optional, default: "FALSE"]
  # When adding new comments on pull requests, keep old comments that were
  added by previous scans.
  # JF_AVOID_PREVIOUS_PR_COMMENTS_DELETION: "TRUE"
  # [Optional, default: "TRUE"]
  # Fails the Frogbot task if any security issue is found.
  # JF_FAIL: "FALSE"
  # [Optional]
  # Relative path to a Pip requirements.txt file. If not set, the Python
  project's dependencies are determined and scanned using the project
  setup.py file.
  # JF_REQUIREMENTS_FILE: ""
  # [Optional, Default: "TRUE"]
  # Use Gradle wrapper.
  # JF_USE_WRAPPER: "FALSE"
  # [Optional]
  # Frogbot will download the project dependencies if they're not cached
  locally. To download the
  # dependencies from a virtual repository in Artifactory, set the name of
  the repository. There's no
  # need to set this value, if it is set in the frogbot-config.yml file.
  # JF_DEPS_REPO: ""
  # [Optional]
  # Template for the branch name generated by Frogbot when creating pull
  requests with fixes.
  # The template must include ${BRANCH_NAME_HASH}, to ensure that the
  generated branch name is unique.
  # The template can optionally include the ${IMPACTED_PACKAGE} and
  ${FIX_VERSION} variables.
  # JF_BRANCH_NAME_TEMPLATE:
  "frogbot-${IMPACTED_PACKAGE}-${BRANCH_NAME_HASH}"
  # [Optional]
  # Template for the commit message generated by Frogbot when creating pull
  requests with fixes
  # The template can optionally include the ${IMPACTED_PACKAGE} and
  ${FIX_VERSION} variables.
  # JF_COMMIT_MESSAGE_TEMPLATE: "Upgrade ${IMPACTED_PACKAGE} to
  ${FIX_VERSION}"
  # [Optional]
  # Template for the pull request title generated by Frogbot when creating
  pull requests with fixes.
  # The template can optionally include the ${IMPACTED_PACKAGE} and
  ${FIX_VERSION} variables.
  # JF_PULL_REQUEST_TITLE_TEMPLATE: "[🐸 Frogbot] Upgrade
  ${IMPACTED_PACKAGE} to ${FIX_VERSION}"
  # [Optional, Default: "FALSE"]
  # If TRUE, Frogbot creates a single pull request with all the fixes.
```

```
# If FALSE, Frogbot creates a separate pull request for each fix.
# JF_GIT_AGGREGATE_FIXES: "FALSE"
# [Optional, Default: "FALSE"]
# Handle vulnerabilities with fix versions only
# JF_FIXABLE_ONLY: "TRUE"
# [Optional]
# Set the minimum severity for vulnerabilities that should be fixed and
commented on in pull requests
# The following values are accepted: Low, Medium, High, or Critical
# JF_MIN_SEVERITY: ""
# [Optional, Default: eco-system+frogbot@jfrog.com]
# Set the email of the commit author
# JF_GIT_EMAIL_AUTHOR: ""
# [Optional]
# List of comma-separated(,) email addresses to receive email
notifications about secrets
# detected during pull request scanning. The notification is also sent to
the email set
# in the committer git profile regardless of whether this variable is set
or not.
# JF_EMAIL_RECEIVERS: ""
# [Optional]
# Set the list of allowed licenses
# The full list of licenses can be found in:
# https://github.com/jfrog/frogbot/blob/master/docs/licenses.md
# JF_ALLOWED_LICENSES: "MIT, Apache-2.0"
# [Optional]
# Avoid adding extra info to pull request comments. that isn't related to
the scan findings.
# JF_AVOID_EXTRA_MESSAGES: "TRUE"
# [Optional]
# Add a title to pull request comments generated by Frogbot.
# JF_PR_COMMENT_TITLE: ""
script:
# For Linux / MacOS runner:
- |
getFrogbotScriptPath=$(if [ -z "$JF_RELEASES_REPO" ]; then echo
"https://releases.jfrog.io"; else echo
"${JF_URL}/artifactory/${JF_RELEASES_REPO}"; fi)
curl -fLg
"$getFrogbotScriptPath/artifactory/frogbot/v2/[RELEASE]/getFrogbot.sh" |
sh
./frogbot ${FROGBOT_CMD}
# For Windows runner:
#
# - $getFrogbotScriptPath = $(if
([string]::IsNullOrEmpty($env:JF_RELEASES_REPO)) {
"https://releases.jfrog.io" } else {
"$($env:JF_URL)/artifactory/$($env:JF_RELEASES_REPO)" })
# - Invoke-WebRequest -Uri
"$getFrogbotScriptPath/artifactory/frogbot/v2/[RELEASE]/getFrogbot.sh" -
UseBasicParsing | ForEach-Object { & $_.Content }
# - .\frogbot ${FROGBOT_CMD}
```

4.commit code

```
git add .
git commit ‘init’
git push
```

5.config Variables



Runners

Runners are processes that pick up and execute CI/CD jobs for GitLab. What is GitLab Runner?

Artifacts

A job artifact is an archive of files and directories saved by a job when it finishes.

Variables

Variables store information that you can use in job scripts. Each project can define a maximum of 8000 variables. Learn more.

Variables can be accidentally exposed in a job log, or maliciously sent to a third party server. The masked variable feature can help reduce the risk of accidentally exposing variable values, but is not a guaranteed method to prevent malicious users from accessing variables. How can I make my variables more secure?

Variables can have several attributes. Learn more.

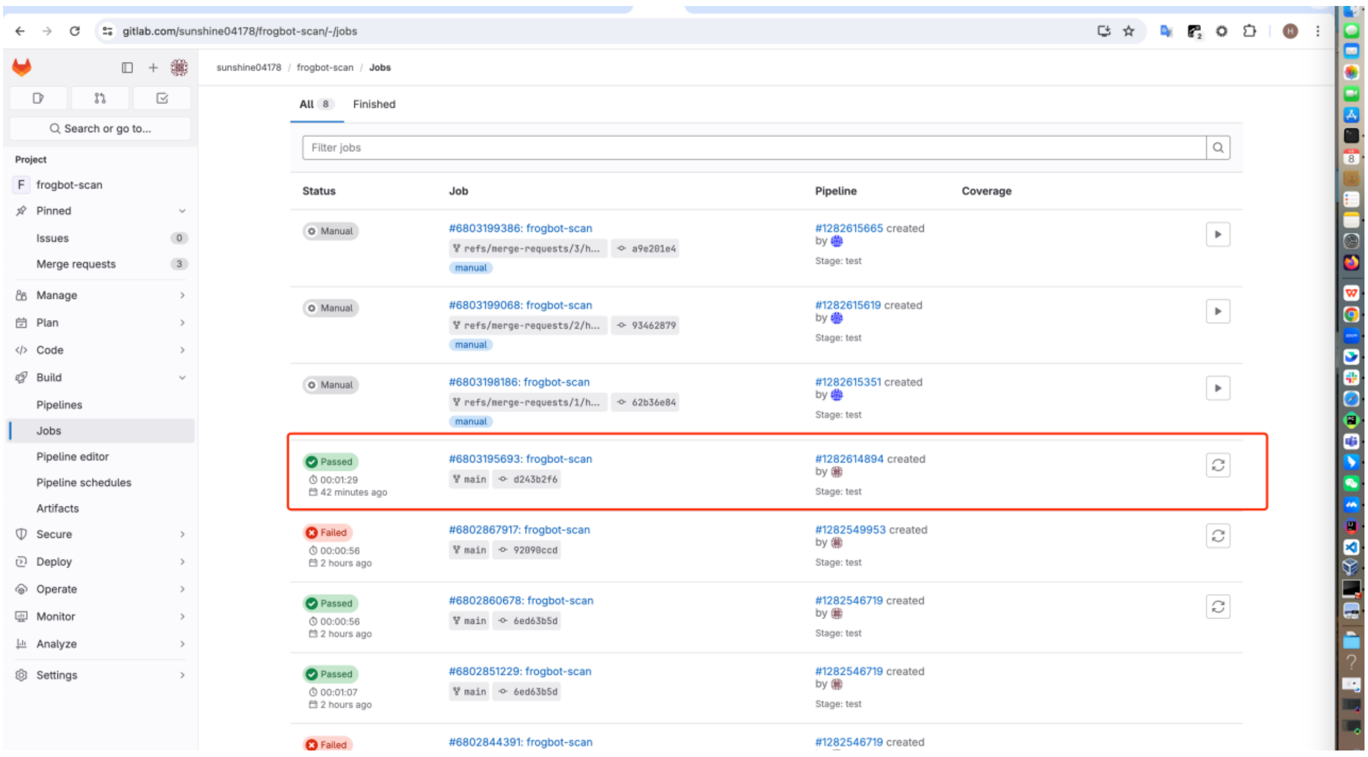
- `Protected:` Only exposed to protected branches or protected tags.
- `Masked:` Hidden in job logs. Must match masking requirements.
- `Expanded:` Variables with `$` will be treated as the start of a reference to another variable.

CI/CD Variables 3

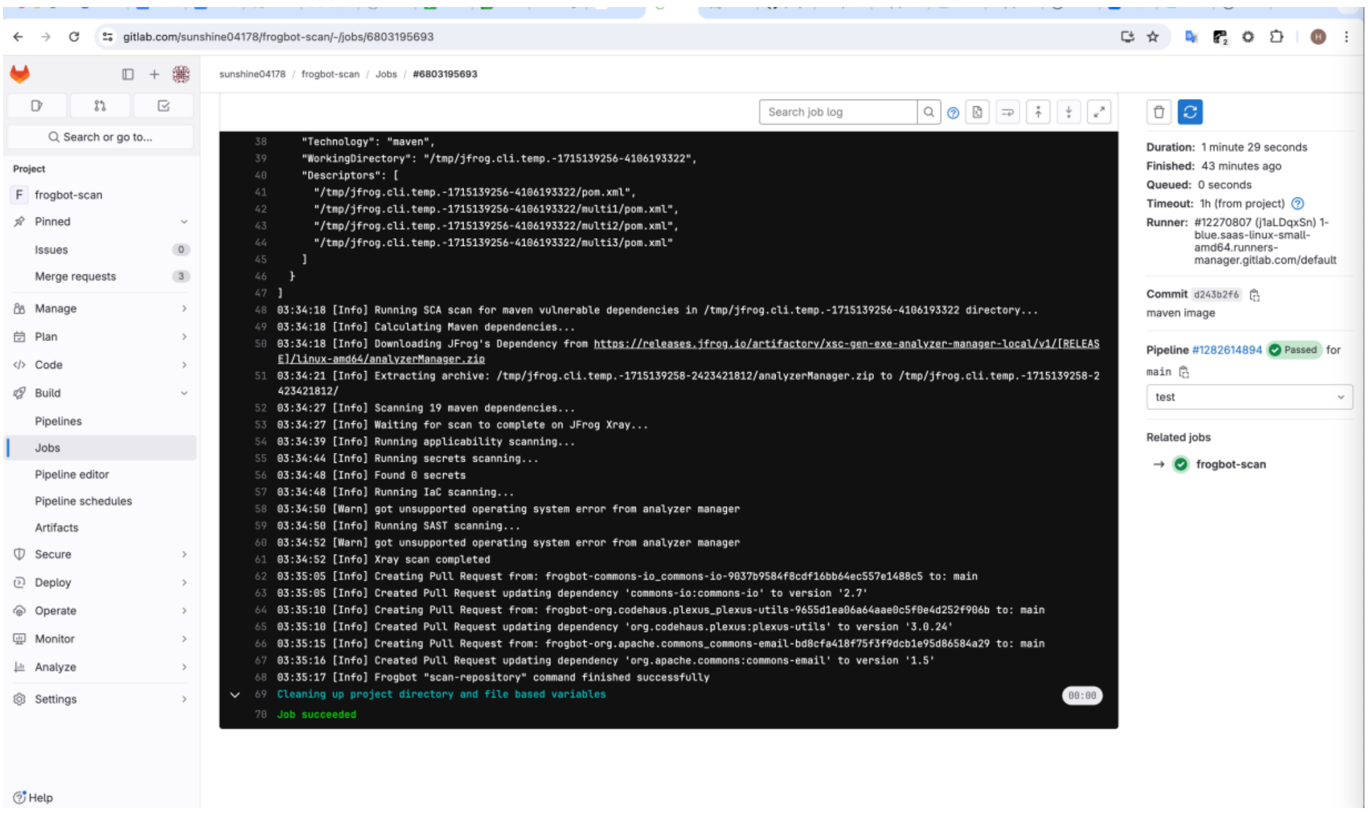
Key ↑	Value	Environments	Actions
JF_ACCESS_TOKEN JF_ACCESS_TOKEN Protected Expanded	*****	All (default)	
JF_URL JF_URL Protected Expanded	*****	All (default)	
USER_TOKEN USER_TOKEN Protected Expanded	*****	All (default)	

Pipeline trigger tokens

6.Run CI



Status	Job	Pipeline	Coverage
Manual	#6803199386: frogbot-scan refs/merge-requests/3/h... a9e201e4 manual	#1282615665 created by Stage: test	
Manual	#6803199068: frogbot-scan refs/merge-requests/2/h... 93462879 manual	#1282615619 created by Stage: test	
Manual	#6803198186: frogbot-scan refs/merge-requests/1/h... 62b36e84 manual	#1282615351 created by Stage: test	
Passed 00:01:29 42 minutes ago	#6803195693: frogbot-scan main d243b2f6	#1282614894 created by Stage: test	
Failed 00:00:56 2 hours ago	#6802867917: frogbot-scan main 92090ccd	#1282549953 created by Stage: test	
Passed 00:00:56 2 hours ago	#6802860678: frogbot-scan main 6ed63b5d	#1282546719 created by Stage: test	
Passed 00:01:07 2 hours ago	#6802851229: frogbot-scan main 6ed63b5d	#1282546719 created by Stage: test	
Failed	#6802844391: frogbot-scan	#1282546719 created	



```
    "Technology": "maven",
    "WorkingDirectory": "/tmp/jfrog.cli.temp.-1715139256-4106193322",
    "Descriptors": [
      "/tmp/jfrog.cli.temp.-1715139256-4106193322/pom.xml",
      "/tmp/jfrog.cli.temp.-1715139256-4106193322/multi1/pom.xml",
      "/tmp/jfrog.cli.temp.-1715139256-4106193322/multi2/pom.xml",
      "/tmp/jfrog.cli.temp.-1715139256-4106193322/multi3/pom.xml"
    ]
  }
]
03:34:18 [Info] Running SCA scan for maven vulnerable dependencies in /tmp/jfrog.cli.temp.-1715139256-4106193322 directory...
03:34:18 [Info] Calculating Maven dependencies...
03:34:18 [Info] Downloading JFrog's Dependency from https://releases.jfrog.io/artifactory/xsc-gen-exe-analyzer-manager-local/v1/[RELEAS
E]/linux-amd64/analyzerManager.zip
03:34:21 [Info] Extracting archive: /tmp/jfrog.cli.temp.-1715139258-2423421812/analyzerManager.zip to /tmp/jfrog.cli.temp.-1715139258-2
423421812/
03:34:27 [Info] Scanning 19 maven dependencies...
03:34:27 [Info] Waiting for scan to complete on JFrog Xray...
03:34:39 [Info] Running applicability scanning...
03:34:44 [Info] Running secrets scanning...
03:34:48 [Info] Found 0 secrets
03:34:48 [Info] Running IaC scanning...
03:34:50 [Warn] got unsupported operating system error from analyzer manager
03:34:50 [Info] Running SAST scanning...
03:34:52 [Warn] got unsupported operating system error from analyzer manager
03:34:52 [Info] Xray scan completed
03:35:05 [Info] Creating Pull Request from: frogbot-commons-io_commons-io-9037b9584f8cdf16bb64ec557e1488c5 to: main
03:35:05 [Info] Created Pull Request updating dependency 'commons-io:commons-io' to version '2.7'
03:35:10 [Info] Creating Pull Request from: frogbot-org.codehaus.plexus_plexus-utils-9655d1ea06a64aae0c5f0e4d252f906b to: main
03:35:10 [Info] Created Pull Request updating dependency 'org.codehaus.plexus:plexus-utils' to version '3.0.24'
03:35:15 [Info] Creating Pull Request from: frogbot-org.apache.commons_commons-email-bd8cfa418f75f3f9dcb1e95d86584a29 to: main
03:35:16 [Info] Created Pull Request updating dependency 'org.apache.commons:commons-email' to version '1.5'
03:35:17 [Info] Frogbot "scan-repository" command finished successfully
Cleaning up project directory and file based variables                                                        00:00
Job succeeded
```

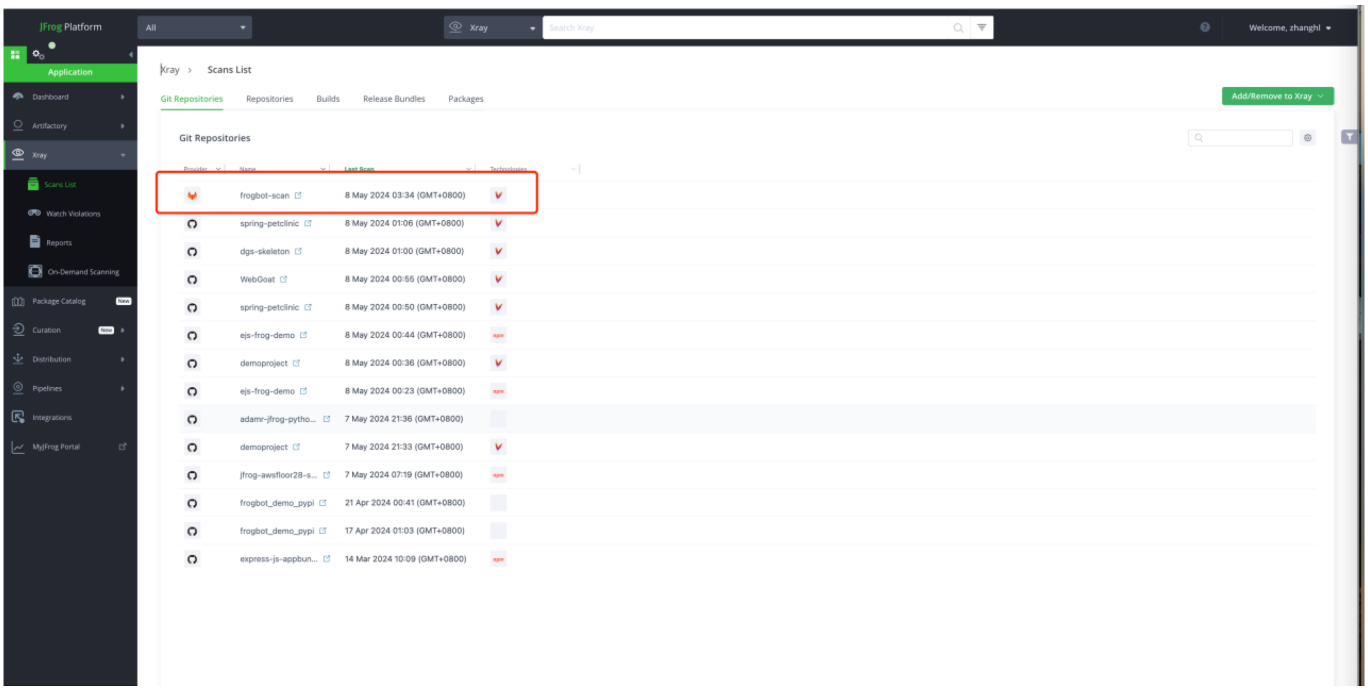
Duration: 1 minute 29 seconds
Finished: 43 minutes ago
Queued: 0 seconds
Timeout: 1h (from project)
Runner: #12270807 (j1aLDqxSn) 1-blue.saas-linux-small-amd64.runners-manager.gitlab.com/default

Commit d243b2f6
maven image

Pipeline #1282614894 Passed for main
test

Related jobs
→ frogbot-scan

7.Check result

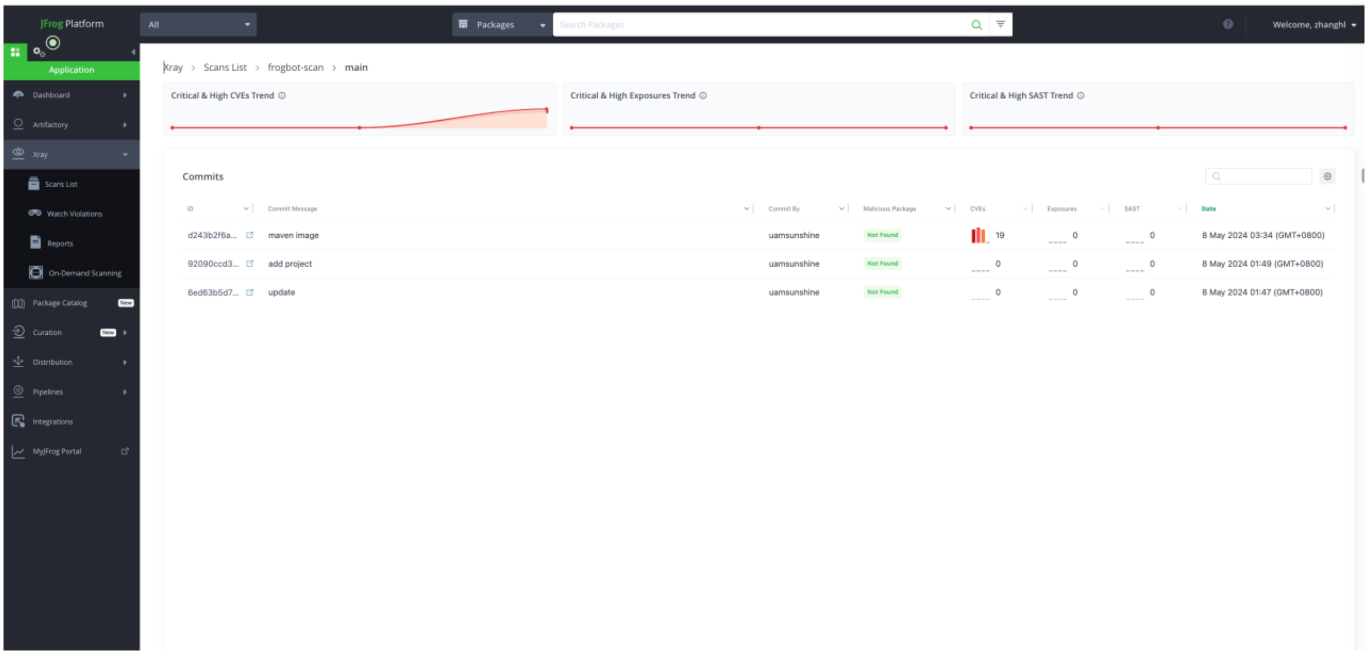


Xray > Scans List

Git Repositories | Repositories | Builds | Release Bundles | Packages

Git Repositories

Provider	Name	Last Scan	Technologies
	frogbot-scan	8 May 2024 03:34 (GMT+0800)	
	spring-petclinic	8 May 2024 01:06 (GMT+0800)	
	dgs-skeleton	8 May 2024 01:00 (GMT+0800)	
	WebGoat	8 May 2024 00:55 (GMT+0800)	
	spring-petclinic	8 May 2024 00:50 (GMT+0800)	
	ejs-frog-demo	8 May 2024 00:44 (GMT+0800)	npm
	demoproject	8 May 2024 00:36 (GMT+0800)	
	ejs-frog-demo	8 May 2024 00:23 (GMT+0800)	npm
	adamr-jfrog-pytho...	7 May 2024 21:36 (GMT+0800)	
	demoproject	7 May 2024 21:33 (GMT+0800)	
	jfrog-awsfloor28-s...	7 May 2024 07:19 (GMT+0800)	npm
	frogbot_demo_pypi	21 Apr 2024 00:41 (GMT+0800)	
	frogbot_demo_pypi	17 Apr 2024 01:03 (GMT+0800)	
	express-js-appbun...	14 Mar 2024 10:09 (GMT+0800)	npm



Xray > Scans List > frogbot-scan > main

Commits

ID	Commit Message	Commit By	Malicious Package	CVEs	Exposures	SAST	Date
d243b2f6a...	maven image	uamsunshine	Not Found	19	0	0	8 May 2024 03:34 (GMT+0800)
92090ccd3...	add project	uamsunshine	Not Found	0	0	0	8 May 2024 01:49 (GMT+0800)
6ed63b5d7...	update	uamsunshine	Not Found	0	0	0	8 May 2024 01:47 (GMT+0800)