

Requirements

Set up an environment with:

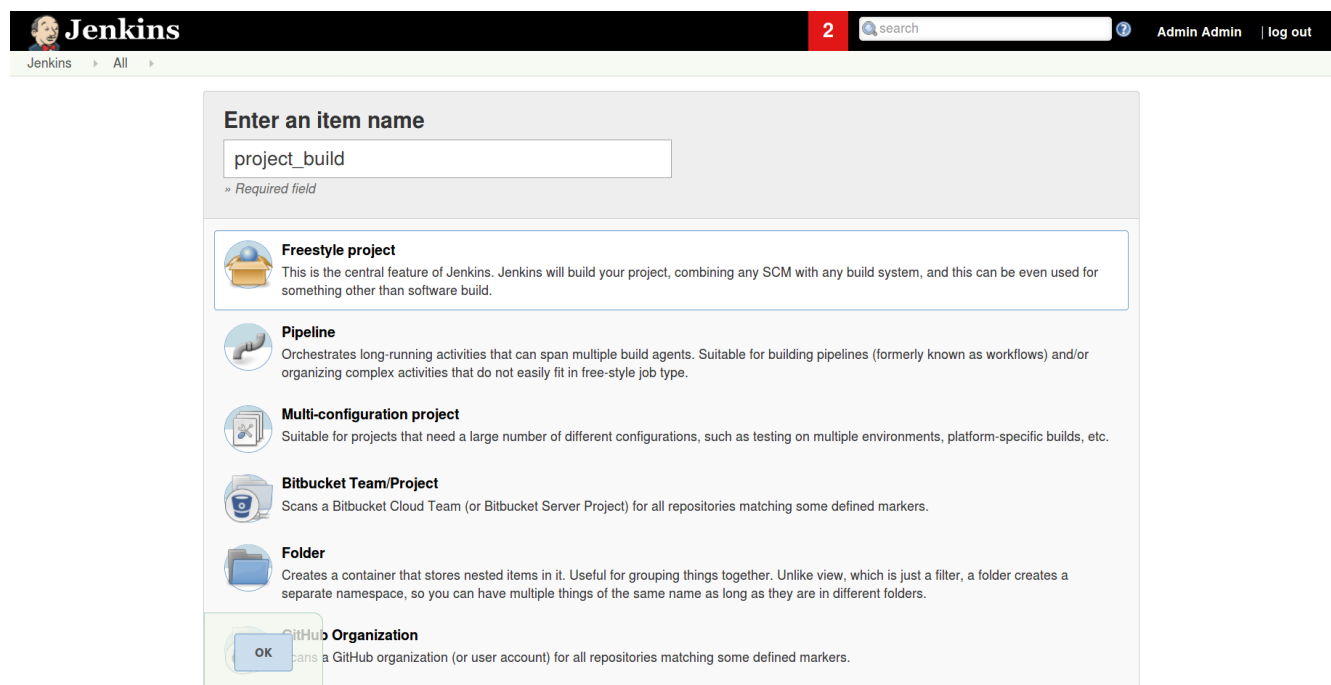
- Jenkins with Gerrit-Trigger plugin installed and configured
(Docs: <https://wiki.jenkins.io/display/JENKINS/Gerrit+Trigger#GerritTrigger-Gerritaccessrights>)
- Gerrit
- FOSSID WebApp

Install FOSSID plugin for Jenkins

From Jenkins > Manage Jenkins > Manage Plugins > Advanced tab> Upload plugin file

“FOSSID.hpi” and restart Jenkins. Copy the configuration file “fossid.properties” to an accessible location (ex. /var/jenkins_home/fossid.properties) and edit it according to your use case,

Create Jenkins job with FOSSID build step:



The screenshot shows the Jenkins 'New Item' page. At the top, the Jenkins logo and navigation bar are visible. The main content area has a header 'Enter an item name' with a text input field containing 'project_build'. Below the input field is a red asterisk and the text 'Required field'. Underneath, there are several project type options, each with an icon and a description:

- Freestyle project**: This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
- Pipeline**: Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**: Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Bitbucket Team/Project**: Scans a Bitbucket Cloud Team (or Bitbucket Server Project) for all repositories matching some defined markers.
- Folder**: Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- GitHub Organization**: Scans a GitHub organization (or user account) for all repositories matching some defined markers.

The 'Freestyle project' option is highlighted with a blue border. At the bottom left, there is a blue button labeled 'OK'.

In the Build Triggers area select Gerrit Trigger

Jenkins > project_build >

General Source Code Management Build Triggers **Gerrit Trigger** Build Environment Build Post-build Actions

Choose a Server Advanced...

Silent Mode ☐

Trigger on

Patchset Created X ?

Exclude Drafts ☐

Exclude Trivial Rebase ☐ ?

Exclude No Code Change ☐ ?

Add ▼

Dynamic Trigger Configuration ☐ ?

Gerrit Project X ?

Type	Pattern	Type	Pattern
Plain	prj1	Path	** X ?

Add Branch

Add Topic

Save Apply

Click on add "Build step" and select FOSSID:

Jenkins > project_build >

General Source Code Management Build Triggers Gerrit Trigger **Build Environment** Build Post-build Actions

Build Environment

☐ Delete workspace before build starts

☐ Use secret text(s) or file(s) ?

☐ Abort the build if it's stuck

☐ Add timestamps to the Console Output

☐ Generate Release Notes

☐ With Ant ?

Build

Add build step ▼

- Execute Windows batch command
- Execute shell
- FOSSID Plugin
- Invoke Ant
- Invoke Gradle script
- Invoke top-level Maven targets
- JIRA: Add related environment variables to build
- JIRA: Create new version
- JIRA: Issue custom field updater
- JIRA: Mark a version as Released
- JIRA: Progress issues by workflow action

Page generated: Apr 16, 2018 1:47:39 PM GMT [REST API](#) [Jenkins ver. 2.107.1](#)

Give a name and set correct path the configuration file (.properties Java file).

FOSSID Plugin X ?

Name

configFilePath

The workflow and the configuration file fossid.properties:

We will assume 3 servers:

- server A has Jenkins installed
- server B has Gerrit installed and the git repo
- server C has FOSSID WebApp installed

When users will push changes to Git repo (server B) an event will be triggered to server A, which will initiate the job on Jenkins server. The FOSSID Plugin will communicate with the FOSSID WebApp (server C) and will check if there is a project and scan created. If there is not, it will try to create it. Prior to creating the actual scan a target path (on server C) will be created, which will be passed as argument to FOSSID API when creating the scan and the patch file from Gerrit will be copied to the target path. Scan will be executed on server C and the results will be sent by the FOSSID plugin (Server A) to Gerrit (server B).

Use case 1

In the use case 1 for each commit in Gerrit a new scan is created in FOSSID WebApp. The scan name is based on \$GERRIT_PROJECT and \$GERRIT_CHANGE_ID. Also the target path will have this structure: /some/path/\$GERRIT_PROJECT/\$GERRIT_CHANGE_ID. This path will be used for copying files and will be fed as target_path when creating the scan in FOSSID WebApp.

The operations related to creating target path, download patch file from Gerrit and copying it to target path, send feedback to Gerrit will be executed by scripts outside of the FOSSID plugin which are mentioned in the configuration file fossid.properties.

Below you can see how the default fossid.properties look like:

```
api_url = http://YOUR_FOSSID_APP/webapp/api.php
api_user = some_user
api_key = some_password
use_case = 1
# Number of times FOSSID Jenkins plugin will check if the scan finished on FOSSID WebApp
fossid_scan_number_of_tries = 100
# Number of seconds before calls to FOSSID WebApp in order to check for finished scanning.
fossid_scan_wait_time = 10
# A command or a script that creates the folder structure based on $GERRIT_PROJECT and $GERRIT_CHANGE_ID
cmd_create_target_path = ssh mysuser@fossid_server mkdir -p /some_path/$GERRIT_PROJECT/$GERRIT_CHANGE_ID
# Target path that will be used when creating the scan using the FOSSID API
target_path = /some_path/$GERRIT_PROJECT/$GERRIT_CHANGE_ID
# A command or script which copies the pathset from Gerrit to the target path
# Sugestion for how to download patchset from Gerrit: curl -o /var/jenkins_home/downloads/$GERRIT_PATCHSET_REVISION.zip
http://gerrit:8080/changes/$GERRIT_CHANGE_NUMBER/revisions/$GERRIT_PATCHSET_REVISION/patch?zip
cmd_copy_patch_files_to_target_path = /var/jenkins_home/my_special_script.sh
# A command or script which returns FOSSID result to Gerrit. It will receive from FOSSID Jenkins plugin the variable $FOSSID_SCAN_RESULT
cmd_send_feedback_to_gerrit = ssh -p 29418 myuser@gerrit_server gerrit review -p $GERRIT_PROJECT -m $FOSSID_SCAN_RESULT
$GERRIT_PATCHSET_REVISION
```

Use case 2

In the use case 2 there is just one scan created in FOSSID WebApp for each branch found in Gerrit. The scan name is based on \$GEERIT_PROJECT and \$GERRIT_BRANCH environment variables. Also the target path will have this structure: /some/path/\$GERRIT_PROJECT/\$GERRIT_BRANCH. This path will be used for copying files and will be fed as target_path when creating the scan in FOSSID WebApp. After each commit in Gerrit the FOSSID Jenkins plugin will be triggered, the new patchset file will be copied to the target path and FOSSID scan will be executed again with delta only option.

```
api_url = http://YOUR_FOSSID_APP/webapp/api.php
api_user = some_user
api_key = some_password
use_case = 2
# Number of times FOSSID Jenkins plugin will check if the scan finished on FOSSID WebApp
fossid_scan_number_of_tries = 100
# Number of seconds before calls to FOSSID WebApp in order to check for finished scanning.
fossid_scan_wait_time = 10
# A command or a script that creates the folder structure based on $GERRIT_PROJECT and $GERRIT_BRANCH
cmd_create_target_path = ssh mysuser@fossid_server mkdir -p /some_path/$GERRIT_PROJECT/$GERRIT_BRANCH
# Target path that will be used when creating the scan using the FOSSID API
target_path = /some_path/$GERRIT_PROJECT/$GERRIT_BRANCH
....
```

In both use cases the result of the scan made by FOSSID WebApp will be displayed in Gerrit:

Author Administrator <admin@example.com> Apr 16, 2018 4:11 PM
Committer Administrator <admin@example.com> Apr 16, 2018 4:11 PM
Commit 20cd49b4f734cd0e175eb92da61eed099406b0dc (browse)
Parent(s) d03f621f91a94aefb5d37544fda322971aff0dbf (browse)
Change-Id I6b4054d1561baa9e1a6dc1d6073207f5788777ec

Files
Open All Diff against: Base
File Path Comments Size
Commit Message
A special3/BluetoothGattDescriptor.java 268
A special3/BluetoothGattIncludedService.java 110
A special3/BluetoothGattServerCallback.java 160
+538, -0

History
Expand All Hide tagged comments
Administrator Uploaded patch set 1. 4:11 PM
jenkins Patch Set 1: Build Started http://localhost:8080/job/fossid9/1/ 4:11 PM
jenkins 4:11 PM
Patch Set 1:
{12904240:{mirror:IGwRIaJPntKtd8A8uNgXzw,artifact_license:N/A,created:2018-04-16 15:11:54,author:garys-esri,underlying_licenses:Apache-2.0 (39),local_path:20cd49b.diff,version:master,file_license:Apache-2.0,url:https://github.com/garys-esri/arcgis-runtime-samples-java/archive/master.zip,scan_file_id:1246696,artifact:arcgis-runtime-samples-java,hits:41,file:arcgis-runtime-samples-java-master/src/main/java/com/esri/sampleviewer/samples/symbology/PictureMarkerSymbolSample.java,size:8820,file_id:323427,match_type:partial,id:12904240,scan_id:173,updated:2018-04-16 15:11:54},12904241:{mirror:q8jHYV6eSAFjVbnK2dx160,artifact_license:N/A,created:2018-04-16 15:11:54,author:garbelini,underlying_licenses:Apache-2.0 (7),local_path:20cd49b.diff,version:master,file_license:Apache-2.0,url:https://github.com/garbelini/ci-bayes/archive/master.zip,scan_file_id:1246696,artifact:ci-bayes,hits:41,file:ci-bayes-master/ci-bayes/src/main/java/com/enigmastation/extractors/impl/PorterStemmer.java,size:21334,file_id:323427,match_type:partial,id:12904241,scan_id:173,updated:2018-04-16 15:11:54},12904244:{mirror:q8jHYV6eSAFjVbnK2dx160,artifact_license:Apache-2.0,created:2018-04-16 15:11:54,author:,underlying_licenses:,local_path:20cd49b.diff,version:2.0.0.RELEASE,file_license:Apache-2.0,url:https://repo1.maven.org/maven2/com/enigmastation/ci-bayes/2.0.0.RELEASE-sources.jar,scan_file_id:1246696,artifact:ci,hits:41,file:com/enigmastation/extractors/impl/PorterStemmer.java,size:21334,file_id:323427,match_type:partial,id:12904244,scan_id:173,updated:2018-04-16 15:11:54},12904245:{mirror:q8jHYV6eSAFjVbnK2dx160,artifact_license:Apache-2.0,created:2018-04-16 15:11:54,author:,underlying_licenses:,local_path:20cd49b.diff,version:1.0.8,file_license:Apache-2.0,url:https://repo1.maven.org/maven2/com/enigmastation/ci-bayes/1.0.8/ci-bayes-1.0.8-sources.jar,scan_file_id:1246696,artifact:ci,hits:41,file:com/enigmastation/extractors/impl/PorterStemmer.java,size:21334,file_id:323427,match_type:partial,id:12904245,scan_id:173,updated:2018-04-16 15:11:54},12904242:{mirror:IGwRIaJPntKtd8A8uNgXzw,artifact_license:N/A,created:2018-04-16 15:11:54,author:Esri,underlying_licenses:Apache-2.0 (39),local_path:20cd49b.diff,version:master,file_license:Apache-2.0,url:https://github.com/Esri/arcgis-runtime-samples-java/archive