

ORAN Base Docker Images for CI Builds

- [Builder Images](#)
 - [Image bldr-alpine3](#)
 - [Image bldr-ubuntu18-c-go](#)
- [Alpine Library Distribution Images](#)
 - [Image bldr-alpine3-mdclog](#)
 - [Image bldr-alpine3-rmr](#)
- [Deprecated Images](#)
 - [Image bldr-alpine3-go](#)
 - [Image bldr-ubuntu16-c-go](#)

This page lists the available Docker images that have been preconfigured with third-party build tools such as C and Golang compilers, autoconf, make, etc. Image configuration is maintained by files in the "it/dev" Gerrit repository: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=tree;f=bldr-imgs>

These base builder images help meet the following goals:

- Use exactly the same version of third-party libraries by all components.
- Reduce Jenkins build time by avoiding repeats of third-party package installs. Remember that a verify build is performed for every change published to <https://gerrit.o-ran-sc.org>

These base images do not have any RIC libraries. RIC project libraries such as RMR, mdclog, asn1 are published to PackageCloud as binary packages (deb or rpm as appropriate) and should be installed during the build process. However PackageCloud does not support Alpine packages (.apk), so Docker images with Alpine binaries are provided as a workaround.

Most base images are pushed to the LF docker "staging" registry nexus3.o-ran-sc.org:10004 and refreshed weekly. This registry allows regular pushes (overwrites) so that the RIC base images keep up with changes in their respective base images. Some images are pushed to the LF docker "release" registry nexus3.o-ran-sc.org:10002. Browse the available images here: [O-RAN-SC Nexus Registry](#)

But note **THE IMAGE TAGS CHANGE FREQUENTLY** as the contents are extended with new packages and versions. If your build stops working, use the link below to find the latest tag and switch to that. If your team depends on these images, please add yourself as a watcher to this page by clicking on the WATCHING link at the top/right!

Projects can use build images in a Dockerfile "FROM" command in the docker build process. For example, to use the Ubuntu 18 base image (see below for the current tag):

```
FROM nexus3.o-ran-sc.org:10004/o-ran-sc/bldr-ubuntu18-c-go:TAG AS project-build
```

Projects can use library images in a Dockerfile "COPY" command in the docker build process. For example, to copy a file from an Alpine library image (see below for the current tag):

```
COPY --from=nexus3.o-ran-sc.org:10002/o-ran-sc/bldr-alpine3-mdclog:TAG /usr/local/lib/libmdc* /usr/local/lib/
```

Builder Images

These images have many build tools and are appropriate for compiling C and Golang software with a minimal startup cost.

Image bldr-alpine3

This provides an Alpine v3.11 base with C, C++, Golang 1.13.4 and Python 3.7.x. No RIC libraries are included; i.e., no MDCLOG, RMR, etc. For complete details see:

- Dockerfile: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3/Dockerfile>
- Tag file: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3/container-tag.yaml>

The image coordinates are:

- Repository: nexus3.o-ran-sc.org:10004/o-ran-sc/bldr-alpine3
- Tag: 12-a3.11

Image bldr-ubuntu18-c-go

This provides an Ubuntu 18.04 base with C, C++, Golang 1.12.17, 1.13.12 and 1.14.4; go-acc and sonar-scanner-cli. No RIC libraries are included; i.e., no MDCLOG, RMR, etc.. For complete details see:

- Dockerfile: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-ubuntu18-c-go/Dockerfile>
- Tag file: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-ubuntu18-c-go/container-tag.yaml>

The image coordinates are:

- Repository: nexus3.o-ran-sc.org:10004/o-ran-sc/bldr-ubuntu18-c-go
- Tag: 9-ubuntu18.04

Alpine Library Distribution Images

These images provide header files and shared-object libraries for Alpine. These exist purely as workarounds for missing APK support at [PackageCloud.io](https://packagecloud.io). These images provide no build tools. Also, unlike the builder images described above, these images are in the Nexus3 **release** registry (not the staging registry).

Image bldr-alpine3-mdclog

This provides MDCLog header files and shared-object libraries for Alpine. For complete details see:

- Dockerfile: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3-mdclog/Dockerfile>
- Tag file: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3-mdclog/container-tag.yaml>

Docker build files can install the shared object file from this image with this command:

```
COPY --from=nexus3.o-ran-sc.org:10002/o-ran-sc/bldr-alpine3-mdclog:TAG /usr/local/lib/libmdclog.so /usr/local/lib/libmdclog.so
```

The image coordinates are:

- Repository: nexus3.o-ran-sc.org:10002/o-ran-sc/bldr-alpine3-mdclog
- Tags: 0.0.4

Image bldr-alpine3-rmr

These images provides RMR header files and shared-object libraries for Alpine. An image exists for (most of) the released versions of this library. For complete details see:

- Dockerfile: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3-rmr/Dockerfile>
- Tag file: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3-rmr/container-tag.yaml>

Docker build files can install header files from this image with this command:

```
COPY --from=nexus3.o-ran-sc.org:10002/o-ran-sc/bldr-alpine3-rmr:TAG /usr/local/include/rmr /usr/local/include/rmr/
```

Docker build files can install the shared object file from this image with this command:

```
COPY --from=nexus3.o-ran-sc.org:10002/o-ran-sc/bldr-alpine3-rmr:TAG /usr/local/lib64/librmr_si.so /usr/local/lib64/librmr_si.so
```

The image coordinates are:

- Repository: nexus3.o-ran-sc.org:10002/o-ran-sc/bldr-alpine3-rmr
- Tags: 4.1.2, 4.0.5, 4.0.2, 4.0.0, 3.8.2, 3.8.0

Deprecated Images

The following images may still be available in the Nexus registry but are no longer recommended for use.

Image bldr-alpine3-go

Deprecated: Teams that need Alpine should use the bldr-alpine3 image. Teams that need RMR should pull that library from a bldr-alpine3-rmr image.

This provides an Alpine v3.11 base with Golang 1.13.8; also the NNG and RMR v3 (SI95) libraries. As of 10 April 2020 the NNG library is no longer included. For complete details see:

- Dockerfile: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3-go/Dockerfile>
- Tag file: <https://gerrit.o-ran-sc.org/r/gitweb?p=it/dev.git;a=blob;f=bldr-imgs/bldr-alpine3-go/container-tag.yaml>

The image coordinates are:

- Repository: nexus3.o-ran-sc.org:10004/bldr-alpine3-go
- Tag: 6-a3.11-rmr3

Image bldr-ubuntu16-c-go

Deprecated: Teams that need Ubuntu should use the 18.04 version, see below.

This provides an Ubuntu 16.04 base with C, C++, Golang 1.12.0 and NNG version 1.1.1. For complete details see:

- Dockerfile: <https://gerrit.o-ran-sc.org/r/gitweb?p=ci-management.git;a=blob;f=docker/bldr-ubuntu16-c-go/Dockerfile>
- Tag file: <https://gerrit.o-ran-sc.org/r/gitweb?p=ci-management.git;a=blob;f=docker/bldr-ubuntu16-c-go/container-tag.yaml>

The image coordinates are:

- Repository: nexus3.o-ran-sc.org:10004/bldr-ubuntu16-c-go
- Tag: 3-ubuntu16.04-nng