

DCAE Committer Guidelines

- [Review window](#)
- [Configure Gerrit watch for auto-notification](#)
- [Check the Commit Message:](#)
- [Check the License Blocks and Copyright Lines on all code and documentation files](#)
- [Check for comments in all new code](#)
- [Check the code](#)
- [Check whether the version number should be updated](#)
- [Check the Build Console Output](#)
- [Were there any -1 or -2 on previous patches by another committer or the PTL?](#)
- [Self-Release yaml](#)
- [ChangeLog.md file](#)
- [Vacation Notice](#)
- [onap-gerrit-review information](#)

In addition to the guidelines specified [Committer Best Practices](#), all DCAE committers are expected to look at these areas before merging a patch. If any item is not followed, you should reply with a -1. Also add comments in appropriate places within the code.

Many of the following checks have been incorporated into the `onap-gerrit-review` tool and will be test**

Review window

- Avoid same day approval unless change is fixing an broken job or critical bug.
- As committers are spread across different timezone, provide **at least a 24 hour window** before merging any non-critical updates.
- Include other committers (if not already added into review) and primary contributors for reviews.

Configure Gerrit watch for auto-notification

- Go to Gerrit SettingsNotification to setup auto email alerts to receive notification on submission into DCAE repository. This will help when submitter miss to add explicitly all the committers for reviews.
- Also check the Gerrit [dashboard](#) for yourself.

Check the Commit Message:

- Is the JIRA ticket issue related to this fix?
- Do either the issue or the commitment message properly describe what is being changed and why?
- For self-release yaml, ensure comment includes a description of changes/bug fixes introduced in that version. If multiple jira's are consolidated - include all Jira references and a brief summary.

Check the License Blocks and Copyright Lines on all code and documentation files

- `onap-gerrit-review`: Is there a `LICENSE.txt` file?
- `onap-gerrit-review`: All code modules should have comments at the beginning that look like this, using the appropriate comment convention:

```
# =====
# Copyright (c) 2017-2020 Company1. All rights reserved.
# Copyright (c) 2020 Company2. All rights reserved.
# Copyright (c) 2020 Company3. All rights reserved.
# =====
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
# =====LICENSE_END=====
```

- All documentation files should have comments at the beginning that look like:

```
.. This work is licensed under a Creative Commons Attribution 4.0 International License.  
.. http://creativecommons.org/licenses/by/4.0  
.. Copyright (c) 2017-2020 Company1. All rights reserved.
```

- onap-gerrit-review: ~~Do they mention the current year for the company doing the modification?~~
- onap-gerrit-review: ~~Note that there is no separator between Copyright lines by different companies.~~
- onap-gerrit-review: ~~Note that the word "Copyright" is capitalized.~~
- onap-gerrit-review: ~~Note that when a company updates code that was previously copyrighted by them, the date range should be extended as shown.~~
- onap-gerrit-review: ~~There is no alternate wording for the copyright lines, such as "Modifications Copyright"~~
- onap-gerrit-review: ~~Do ALL of the copyright notices for the current gerrit push document the same company?~~

Check for comments in all new code

We should improve the documentation in the code whenever possible. At a minimum, NEW code should be documented.

- javadocs, pydocs format is preferred on all public methods, classes and modules (files)

Check the code

- Most importantly, does it actually fix what the commitment message says should be fixed?
- Verify against ONAP code standards. In general, we use the [Google Style Guide](#) with some small modifications.
 - Java style is specified [here](#), which points to the [Google Style Guide](#) repo, but with indentation level of 4 and a column limit of 120.
 - Javascript code should meet the similar [Javascript Google Style Guide](#), except for the number of characters in one line of code is not restricted.
 - Python code should similarly match the [Python Google Style Guide](#), but with a column limit of 120.
 - C++ and C code should similarly match the [C++ Google Style Guide](#), but with a column limit of 120.
- Did the unit tests get updated?
 - New functionality MUST have unit tests.
 - New methods added to existing code MUST be covered by existing or additional unit tests.
 - We are looking for a minimum of 60% code coverage, with higher levels encouraged.
- Using the knowledge of Security design principles, does the code meet the security requirement and not introduce any vulnerabilities.
- Verify that no non-encrypted credentials (non-test) are stored in the repositories

Check whether the version number should be updated

- onap-gerrit-review: ~~if there is any new feature/enhancement:~~
 - ~~the patch version should be bumped~~
- onap-gerrit-review: ~~if a change is a bug fix on a previously merged patch, AND if that previous version is not already released:~~
 - ~~then changing the version is optional~~
- The repos need to have the version number expressed in multiple places
 - onap-gerrit-review: ~~In pom.xml, the project/version value is always specified as either W.X.Y or W.X.Y-SNAPSHOT, as in 1.3.2-SNAPSHOT.~~
 - onap-gerrit-review: ~~Every directory that has a pom.xml file should ALSO have a version.properties file AND a ChangeLog.md file.~~
 - ~~The exception is when there are subdirectories with individual pom.xml files. In that case, those directories should have a ChangeLog.md file. Typically these match the docker containers.~~
 - onap-gerrit-review: ~~The same value (without any "-SNAPSHOT" suffix) will be specified in version.properties, separated out into separate major, minor and patch values:~~
 - ~~major=W~~
 - ~~minor=X~~
 - ~~patch=Y~~
 - onap-gerrit-review: ~~The same value will be specified in the ChangeLog.md file.~~
 - onap-gerrit-review: ~~These values MUST match:~~
- onap-gerrit-review: ~~When an artifact (e.g docker container) is "released", a corresponding file will be added to the releases directory. The version of that release file must also match the version found in the pom.xml, version.properties and ChangeLog.md files.~~

Check the Build Console Output

- Errors during the build process will cause the verify step to post a -1, preventing merger.
- However, there are a other things to still look for in the build console output.
 - In java and python code, does the "lint" pass look okay?
 - Look for "hidden pieces of coal". Lots of stylistic issues (say about indentation and white space) can hide real issues.
- Are the unit tests running cleanly?
 - For example, you shouldn't see stack traces from unit tests, as the unit tests should capture those.
- Check the code coverage statistics when available.
 - Python code will generally show code coverage statistics in the build console output.
 - Make certain that we don't fall below 60% code coverage. (Numbers closer to 80% are encouraged.)
 - Java code usually does not show in out build logs, but are currently generated after merger.

Were there any -1 or -2 on previous patches by another committer or the PTL?

- This is a FULL STOP.
- Please DO NOT merge the code until:
 - that other committer has given at least a subsequent +1 or
 - the PTL says it is okay to +2 anyway (which would be very rare)

Self-Release yaml

- Ensure no outstanding patch remaining in gerrit for review/merge for container being released. We would want to avoid releasing container too frequently as well (consider consolidating release for multiple patch)
- As artifact release impacts different repositories (blueprint/bootstrap, oom etc); consolidate release request (and subsequent update to other impacted repositories)
- For self-release yaml, ensure comment include description of changes/bug fixes introduced in that version. If multiple jira's are consolidated - include all Jira reference/brief summary

ChangeLog.md file

In the Istanbul release, we have adopted the practices of keeping a changelog. Best practices for changelogs can be found at <keepachangelog.com>.

An example of a Changelog is:

```
# Change Log
All notable changes to this project will be documented in this file.

The format is based on [Keep a Changelog](http://keepachangelog.com/)
and this project adheres to [Semantic Versioning](http://semver.org/).

## [1.0.6] - 2021/08/28
### Changed
- [DCAEGEN2-2885] - DCAE SliceAnalysis MS - CPS Integration
### Removed
- [DCAEGEN2-2811] - Remove security vulnerabilities

. . .
```

As committers, some of the things you should always check are:

- onap-gerrit-review: ~~The name of the changelog may be specified in any case, but at least the letter "C" must be capitalized. We are using markdown, so the extension must always be ".md".~~
- Within the changelog:
 - onap-gerrit-review: ~~The date stamp must always be specified as either YYYY-MM-DD or YYYY/MM/DD.~~
 - onap-gerrit-review: ~~The JIRA ticket associated with the changes that went into a version should be noted.~~
 - It is not necessary to have both the JIRA ticket number AND a link to the JIRA ticket.
 - There should be an entry for every single version.
 - onap-gerrit-review: ~~The latest version must always be first, in reverse time order.~~
 - The date stamp should always reflect the most recent change associated with a given version.
 - An [Unreleased] section is optional. (JIRA tickets should be used in lieu of an [Unreleased] section.)

Vacation Notice

- Notify other committers/PTL on vacation plans
- Set status accordingly in Gerrit (under Gerrit->Setting Profile)
- Update weekly meeting "vacation" section specifying the OOO days/weeks

onap-gerrit-review information

The tool [onap-gerrit-review](#) can be found on Github. It automates many of the checks listed above. It is used automatically by the Jenkins job, but may also be used by committers on code that needs to be reviewed, as well as by code submitters before they submit the code.

If you suggestions for improvements, please provide issues or pull requests on Github.