

# Contribution Guidelines

The repository centralises all the Ansible roles. The roles are all part of the Galaxy. We love contribution and we love giving visibility to our contributors, this is why all the **commits must be signed-off**.

## Mailing list

Please register the mailing list at <http://lists.ceph.com/listinfo.cgi/ceph-ansible-ceph.com>

## IRC

Feel free to join us in the channel `#ceph-ansible` of the OFTC servers

## Github

The main github account for the project is at <https://github.com/ceph/ceph-ansible/>

## Submit a patch

To start contributing just do:

```
$ git checkout -b my-working-branch
$ # do your changes #
$ git add -p
```

If your change impacts a variable file in a role such as `roles/ceph-common/defaults/main.yml`, you need to generate a `group_vars` file:

```
$ ./generate_group_vars_sample.sh
```

You are finally ready to push your changes on Github:

```
$ git commit -s
$ git push origin my-working-branch
```

Worked on a change and you don't want to resend a commit for a syntax fix?

```
$ # do your syntax change #
$ git commit --amend
$ git push -f origin my-working-branch
```

## PR Testing

Pull Request testing is handled by jenkins. All test must pass before your PR will be merged.

All of tests that are running are listed in the github UI and will list their current status.

If a test fails and you'd like to rerun it, comment on your PR in the following format:

```
jenkins test $scenario_name
```

For example:

```
jenkins test luminous-ansible2.3-journal_collocation
```

# Backporting changes

If a change should be backported to a `stable-*` Git branch:

- Mark your PR with the GitHub label “Backport” so we don’t lose track of it.
- Fetch the latest updates into your clone: `git fetch`
- Determine the latest available stable branch: `git branch -r --list "origin/stable-[0-9].[0-9]" | sort -r | sed 1q`
- Create a new local branch for your PR, based on the stable branch: `git checkout --no-track -b my-backported-change origin/stable-3.0`
- Cherry-pick your change: `git cherry-pick -x (your-sha1)`
- Create a new pull request against the `stable-3.0` branch.
- Ensure that your PR’s title has the prefix “backport:”, so it’s clear to reviewers what this is about.
- Add a comment in your backport PR linking to the original (master) PR.

All changes to the stable branches should land in master first, so we avoid regressions.

Once this is done, one of the project maintainers will tag the tip of the stable branch with your change. For example:

```
git checkout stable-3.0
git pull --ff-only
git tag v3.0.12
git push origin v3.0.12
```