

Becoming a contributor

This document provides general information about the ways of contribution to the SONiC Community.

Joining the SONiC Community

SONiC community welcomes new members to join this open source community project and contribute for SONiC development.

1. Please get familiar with our [governance document](#)
2. Members are expected to follow the [code of conduct](#) and agree.
3. Members are welcome to join the [mailing list](#) and start attending the weekly discussion on the ongoing activities to contribute.
4. Members are welcome to join [SONiC community meetings](#). Please reserve the slots in your calendar.
5. Members are welcome to join the [sub work groups](#) and contribute.
6. Members shall go through the [SONiC webpage](#) and [SONiC Wiki](#) and familiarize themselves about the basics.

Legal

We welcome everyone to contribute to SONiC project. Individual Contributor License Agreement (ICLA) is required (Corporation CLA is not supported) before you make a contribution to the SONiC open source project. Please follow the process and sign the ICLA [here](#). You will get the sign link when you submit your first PR. Thanks for your support.

How to Contribute

Following are the way of contributions and its process for the community.

Technical Contribution

Community members or the SONiC partners are welcome to contribute in all of the following development process.

The ways in which the members can contribute are explained in detail in the following sub sections.

1. Development
2. Reviews
3. Testing
4. Bug Fixing
5. Presentation and demos
6. Documentation

Development

Members can contribute by owning and develop a feature for SONiC or enhance an existing feature/sub-features. Members have to inform the community about ownership of feature development/enhancement and ensure that the feature is added to the roadmap. Members should also provide a rough plan about the design document availability date and code availability date in advance and also identify reviewer(s). Members shall do the design, coding & testing by following the process explained in this document and ensure that the feature is merged before the release.

Reviews

Members can contribute by owning up the reviews for the design documents, code, test cases/scripts and any documents that are prepared for SONiC. Members can join the design review meetings and provide review comments during the discussion and/or provide the review comments on the pull request raised by the authors. Members are welcome to complete the review cycle on time so that the features planned in the roadmap are completed before the release.

Testing

Members are welcome to test SONiC in their own platforms and raise issues in the appropriate SONiC repositories. Members can also contribute by reviewing

the existing test cases and adding new test cases to the sonic-mgmt repository.

Bug Fixing

Members are welcome to fix the bugs present in SONiC. They are requested to join the issue triage sub group and contribute for root cause analysis as well as in review.

Presentation and demos

Regular workshops and OCP meetings are happening for SONiC. Members are welcome to participate and contribute for technical presentation and demos that can be used during the workshop. Partners are welcome to set up booths at the OCP summits with their platforms running SONiC.

Documentation

Members are welcome to review the existing documents in SONiC and also help in enhancing the existing technical documents like Config guide, CLI guide, Wiki pages, etc., Members are welcome to capture the SONiC deployment scenarios as well as the use cases in a document and add it to the SONiC Wiki library.

1. High level design documents <https://github.com/sonic-net/SONiC/tree/master/doc>
2. Test plans <https://github.com/sonic-net/sonic-mgmt/tree/master/docs>
3. SONiC release notes <https://github.com/sonic-net/SONiC/tree/master/doc>
4. SONiC command reference <https://github.com/sonic-net/sonic-utilities/blob/master/doc/Command-Reference.md>
5. SONiC YANG models <https://github.com/sonic-net/sonic-buildimage/tree/master/src/sonic-yang-models/yang-models>
6. SONiC YANG model guidelines <https://github.com/sonic-net/SONiC/blob/master/doc/mgmt/SONiC YANG Model Guidelines.md>

7. Technical Charter <https://github.com/sonic-net/SONiC/blob/master/SONiC%20Foundation%20Technical%20Charter.pdf>

Development Process

Design

1. Create the file as MD (markdown) format by using template https://github.com/sonic-net/SONiC/blob/master/doc/hld_template.md;
2. Send as a pull request to <https://github.com/sonic-net/SONiC/pulls>;
3. Go through community design review;
4. Once approved, the doc will be merged to <https://github.com/sonic-net/SONiC> repo;
5. The doc will be published on SONiC Wiki;

Code

1. Code the feature and raise a pull request
2. Pull request for the code should be raised from appropriate repository.
3. Available repository for code are sonic-buildimage, sonic-utilities, sonic-swss, sonic-build-system, etc.,
4. Members should inform the community if a new repository is created the feature.
5. Code has to be in file format such as python, C/C++, ruby, etc.,
6. Go through community for code review.
7. Once approved, the code will be merged to <https://github.com/sonic-net/SONiC> repo.

Repositories

Members should pull a separate private branch from the master branch in an appropriate repository. There are various code repository available such as sonic-buildimage, sonic-utilities, sonic-swss, sonic-build-system etc. Members should ensure that the owned feature is updated in the right repository to add the development code and its corresponding HLD document.

Release Process

1. The community will follow the roadmap for release and would be updated periodically. The community currently schedules two releases annually: the May release (release date: end of May) and the November release (release date: end of Nov). Please monitor <https://lists.sonicfoundation.dev/g/sonic-dev> mail list for the release plan info.
2. The release tracking will be followed for all the ongoing feature on the release and its corresponding HLD discussion will be happening every week.
3. A member should ensure the active involvement in the discussion.
4. The release tracking sheet should be updated periodically to ensure the status of the ongoing release.
5. Release notes will be issued upon successful release for the completed features.
6. Any feature that is not completed before the branching will be considered for next release.
7. For further details about the SONiC release train click [here](#)

Pull Request Process

Find below the steps to raise a Pull Request for the HLD document as well as for the development Code.

1. Fork the repository in which you want to make the changes. For example, if you want to make changes in sonic-buildimage, go to <https://github.com/sonic-net/sonic-buildimage> and click "Fork". If you have already forked that repository, you can either rebase your forked repository (to sync it to the latest repo), or, you can delete your forked repository and fork afresh.
2. Clone your forked repository and go to the base directory
Example: <https://github.com/Your Username/sonic-buildimage.git> cd sonic-buildimage

3. Create a private branch in your forked repository. Use simple meaningful branchnames. Example: `git checkout -b your_branchname`
4. Do "push origin". Example: `git push origin your_branchname`
5. Do necessary changes in the files. You can add new files, or delete old files or modify existing files. Ensure that the code/doc is reviewed (and tested if required) before pushing the changes to repository.
6. Do "git status" to check the list of files that are added/modified/deleted. Ensure that no other unwanted files are shown in the diff.
7. Do "git add ." to add all the changes for commit.
8. Do `git commit -m "your explanation about the change"`. Example: `git commit -m "Added code for CLI commands for management VRF feature"`
9. Push the changes to your forked repository. This will push all your changes to your branch inside your forked repository. Example: `git push --set-upstream origin your_branchname`
10. Go to your repository in the webpage (example: https://github.com/Your_Username/sonic-buildimage) where you will see the option for creating the pull request. It will show the recent push that you had done and it will have an option for "Compare And Pull Request". Click the same and create the pull request.
11. Ensure to add label/tag for the feature raised. example - [PR#2174](#) where, Generic Config and Update feature has been labelled as GCU.
12. Provide all details in the pull request and raise it.
13. Follow-up with community for review. Address review comments and follow-up with repository owners for merging the pull request.
14. Use HLD PR to track all the related code PRs, if no HLD PR, please create a dedicated tracking PR to track the

code PRs. Example - [PR#806](#). Please note: if your feature has new CLI or update existing CLIs, please make sure to update the <https://github.com/sonic-net/sonic-utilities/blob/master/doc/Command-Reference.md>

The below example [PR#806](#) defines the md format followed in updating the HLD PR's within the code PRs. MD file format to be followed in HLD PR for updating the Code PR's :

```
| Repo | PR title | State |
| ----- | ----- | -----
--- |
| sonic-swss | [EVPN VxLAN enhancement to support P2MP
tunnel based programming for Layer2
extension](https://github.com/Azure/sonic-
swss/pull/1858 ) | ![GitHub issue/pull request
detail](https://img.shields.io/github/pulls/detail/state
/Azure/sonic-swss/1858 [img.shields.io]) |
| sonic-buildimage | [Enhancing vs support to mock
based on platform](https://github.com/Azure/sonic-
buildimage/pull/8685) | ![GitHub issue/pull request
detail](https://img.shields.io/github/pulls/detail/state
/Azure/sonic-buildimage/8685 [img.shields.io]) |
| sonic-sairedis | [Added query Enum value capability
for vxlan EVPN feature](https://github.com/Azure/sonic-
sairedis/pull/920 [github.com]) | ![GitHub issue/pull
request
detail](https://img.shields.io/github/pulls/detail/state
/Azure/sonic-sairedis/920 [img.shields.io]) |
| sonic-swss | [VxLAN Tunnel Counters and Rates
implementation](https://github.com/Azure/sonic-
swss/pull/1859 ) | ![GitHub issue/pull request
detail](https://img.shields.io/github/pulls/detail/state
/Azure/sonic-swss/1859 [img.shields.io]) |
```

In the MD file format above,

sonic-swss defines the repo name
EVPN VxLAN enhancement to support P2MP tunnel based
programming for Layer2 extension defines the PR title
and
GitHub issue/pull request detail describes the current
state of the PR

The above MD file format updation will be reflected in HLD PR page as shown below

How to file an Issue ?

1. Issues should be filed under <https://github.com/sonic-net/sonic-buildimage/issues>;
2. Please fill the template as
 - a. Description
 - b. Steps to reproduce the issue:
 - c. Describe the results you received:
 - d. Describe the results you expected:
 - e. Attach debug file
3. Submit
4. Members shall join the Biweekly [Issue Triage meeting](#) and contribute towards raising and resolving the issues.
5. If one issue needs to be linked to one specific branch, please use label "Issue for ", e.g. Issue for 202305

Design Spec

1. Create the file as MD (markdown) format with [HLD template](#)
2. Send as a pull request to <https://github.com/sonic-net/SONiC/pulls>;
3. Go through community design review;
4. Once approved, the doc will be merged to <https://github.com/sonic-net/SONiC> repo;
5. The doc will be published on SONiC WiKi;