Developer Getting Started

Getting started with your account, Gerrit & Git

Linux Foundation ID

For access to OPNFV’s infrastructure, you will need a Linux Foundation account. (Please use your company email address for the Linux Foundation account and don’t have spaces or special characters in your username)

Tutorial video for Linux Foundation ID creation

Participating in a project

To participate, via contribution, in any project in OPNFV, you will need to contact Linux foundation support https://docs.releng.linuxfoundation.org/en/latest/helpdesk.html If you are not from an LF Networking Fund (“LFN”) member company, you will have to agree to be subject to and bound by the OPNFV IP Policy prior to your project contribution. If you’re not from an LFN member company, please visit this page for the authorization from non-member developers.

Once the rights have been set-up you are ready to add your contributions to the project.

Common Tools

There are a number of tools used to communicate amongst OPNFV project teams. While many of these are listed on our www site that list is not 100% complete so we will replicate and update the list here.

Mailing List

The main mailing list is called Technical Discuss.

• Sign up here: https://lists.opnfv.org/mailman/listinfo/opnfv-tech-discuss
• This list is for general technical discussion about OPNFV. It’s open for anyone to join.
• If you are already a member you can click this link to send an email to the list: mailto:opnfv-tech-discuss@lists.opnfv.org
• Other mailing lists are available here: https://www.opnfv.org/developers/tools/mailing-lists
• Topic filters allows you to choose to only receive messages from the mailing list for particular topics (more info. can be found at http://web.mit.edu/lists/mailman/topics.html). For the Technical Discuss mailing list, the following topic filters are available:
  • vspf: Discussion channel for the vswitchperf project
  • sfqm: Discussion channel for the Software Fastpath Service Quality Metrics project
  • doctor: Discussion channel for the Fault Management (Doctor) project
  • copper: Discussion channel for the Virtualized Infrastructure Deployment Policies (Copper) project
  • promise: Discussion channel for the Resource Management (promise) project
  • bgs: Discussion channel for the Bootstrap/Get started (Genesis) project
  • opendaylight: Discussion channel for OpenDaylight-OPNFV interlock
  • availability: Discussion channel for the High Availability for OPNFV (availability) project
  • vnfflash: Discussion channel for the OpenStack Based VNF Forwarding Graph (vnf forwarding_graph) project
  • ocopus: Discussion channel for the Continuous Integration (Octopus) project
  • joid: Discussion channel for the Juju OPNFV Infrastructure Deployer (JOID) project
  • ipv6: Discussion channel for the IPv6-enabled OPNFV (ipv6) project
  • pharos: Discussion channel for the Testbed infrastructure (Pharos) project
  • functest: Discussion channel for the Base system functionality testing (functest) project
  • parser: Discussion channel for the Deployment Template Translation (Parser) project
  • oscar: Discussion channel for the OSCAR (OPNFV System Config & Reporting) project
  • announce: Discussion channel for general announcements
  • opnfv-scc: Discussion channel for the OPNFV Security group
  • infrac: Discussion channel for the OPNFV hardware infrastructure
  • release: Discussion channel for OPNFV release discussions
  • SFC: Discussion channel for OPNFV Service Function Chaining
  • yardstick: Discussion channel for OPNFV Yardstick
  • onoslw: Discussion channel for ONOS Framework project
  • ovn4nf: Discussion channel for the Open vSwitch for NFV project
  • netready: Discussion channel for the NetReady project
  • qtip: Discussion channel for the QTIP project
  • ovn4nf: Discussion channel for the OVN for NFV project
  • bottlenecks: Discussion channel for Bottlenecks project which focuses on stress/tuning/system limit testing
  • armband: Discussion channel for the Armband project
  • (You can select topics you want to subscribe to in the configuration page at https://lists.opnfv.org/mailman/options/opnfv-tech-discuss)

Please subscribe to at least [yourproject] and [announce] Your sysadmin will use [announce] for any downtimes that may be needed. (I’ll post downtime requests to infra as well)
IRC

There are a number of IRC channels used for various OPNFV purposes.

General Purpose and Working Group Channels

- #opnfv-meeting <-- used for most weekly team meetings. It's recommended to not use a specific IRC meeting channel for your meetings. All project meetings should share the main IRC channel to avoid overlapping meetings. One should finish before the next scheduled one starts.
- #opnfv-release <-- used for weekly release meetings and bi-weekly APAC release meeting. Also available for asking questions about the release process and schedule.
- #opnfv-testperf <-- performance testing team
- #opnfv-mano <-- used by mano-wg for cross-project mano integration and best practices compilation

Project Specific Channels

Projects will typically have their own channel to conduct project specific meetings and other communication. The project wiki page will typically list the project's IRC channel. The project IRC channel uses the format #opnfv-project_name>. For example, the IRC channel for the Doctor project is #opnfv-doctor. If you're still unsure, check with the project PTL.

Git

OPNFV uses Git repositories for our code, documentation, and test artifacts. Git is a common tool for open source collaborative projects and there are many resources to help you become a master at working with Git. A good place to start is the Git how to guide.

Our development processes will help guide you how to work with each project type. Once that material is ready we will provide links here for performing various tasks in OPNFV. For now you can get started by browsing the projects on Gerrit and setting up your security. Git-Review as listed below is a great tool for simplifying the basics when working with Git.

Gerrit

Gerrit is our code review and maintenance vehicle, this is where you will go to comment and review existing patches, this is where your patches will show up once you have pushed them to the Git repository. You can subscribe to be notified to any changes to projects you are interested in. Use your Linux Foundation ID to log in to the OPNFV Gerrit site.

https://gerrit.opnfv.org/gerrit/

General Workflow

Once your local repository is set up, the following workflow is recommended:

- Get the latest changes from the repository:

  $ git remote update  
  $ git checkout master  
  $ git pull --ff-only origin master

- Create a topic branch in which you will work, and switch to it:

  $ git branch NAME_OF_THE_TOPIC_BRANCH  
  $ git checkout NAME_OF_THE_TOPIC_BRANCH

You can also use the shorter version:

  $ git checkout -b NAME_OF_THE_TOPIC_BRANCH

The name of the topic branch should be something related to what’s being changed, since it will show up in Gerrit.

When you submit your changes to gerrit with git-review (see: Welcome to the OPNFV developer resources page.), even though they will show the name of the topic branch, they will end up being merged to master. When crafting your commits, you might find Git Commit Good Practice useful.

For more information, you may find the OpenStack Developer's guide and MediaWiki Gerrit Tutorial useful.

How-to’s on setting up git and getting started with Gerrit
Step-by-step instructions and screenshots for getting **Gerrit access** and working with **git**

[mediawiki's site for Git Gerrit](https://wiki.opendaylight.org/view/GIT_Cheat_Sheet)

[Opendaylight's Cheat Sheet and CLI git reference](https://wiki.opendaylight.org/view/OpenDaylight_Controller:Pulling,_Hacking,_and_Pushing_the_Code_from_the_CLI)

Once you have setup git and created your ssh key you can register your SSH public key in Gerrit here:
https://gerrit.opnfv.org/gerrit/

You can also get a password from Gerrit UI and use this instead of SSH key

**Git-review** is a command-line tool for Git / Gerrit. It makes it easier to configure git, to submit a change or to fetch an existing one.

[Opendaylight's page on working with git-review](https://wiki.opendaylight.org/view/OpenDaylight_Controller:Pulling,_Hacking,_and_Pushing_the_Code_from_the_CLI)

Check out the [OPNFV Git/Gerrit page](https://wiki.opendaylight.org/view/OpenDaylight_Controller:Pulling,_Hacking,_and_Pushing_the_Code_from_the_CLI) for a collection of setup suggestions for the OPNFV Source Code Repositories.

A sample procedure for start hacking in Parser Project: [Parser HowTo Start](https://wiki.opendaylight.org/view/OpenDaylight_Controller:Pulling,_Hacking,_and_Pushing_the_Code_from_the_CLI)