How to Commit

This guide highlights the process and responsibilities of committers on Flume project.

Note: Just like contributors, committers must follow the same guidelines for contributions as highlighted in the How to Contribute guide.

Committing patches

The primary responsibility of committers is to commit patches from various contributors. In order to do so, a committer should follow these guidelines:

- **Review the patch:** Follow the How to Contribute guide's section on reviewing code to ensure that you review the code before committing it. Every patch that is committed to the source control must be reviewed by at least one committer first. If you are not ready to accept the patch in the current state, make sure you update the JIRA to 'Cancel Patch'.

- **Make sure patch is attached to the JIRA with license grant**: Before a patch is committed to the source code, the contributor must explicitly attach the patch to the JIRA and grant it the necessary licence for inclusion in Apache works.

- **Commit the patch:** If the patch meets review expectations and is well tested, it can be committed to the source control. Make sure that the patch applies cleanly to the latest revision and if not, request the patch be rebased accordingly. Once ready for commit, the commit message should have the following format:

  ```
  Flume-XXX. Brief description of the problem.

  (Contributor's Name via Committer's Name)
  ```

- **Mark the JIRA resolved:** After the patch has been committed, you should mark the JIRA resolved and ensure that it's `fixVersion` is set to the next release version number. Make sure to thank the contributor in the comment you add while marking the JIRA resolved.

Contributing patches

The Flume project does not distinguish between committers and contributors with respect to contributing patches. Typically, a committer submitting a patch will follow the same process as expected from a regular contributor and have the reviewing committer checkin the submitted change. This procedure is an informal guideline and not a hard policy since at times committers may have to bypass the long drawn process to commit the change in order to fix a broken build, or work through a release etc.