How to Contribute

We always welcome contributions to help make `go-zfs` better. Please take a moment to read this document if you would like to contribute.

Reporting issues

We use [Github issues](#) to track bug reports, feature requests, and submitting pull requests.

If you find a bug:

- Use the GitHub issue search to check whether the bug has already been reported.
- If the issue has been fixed, try to reproduce the issue using the latest `master` branch of the repository.
- If the issue still reproduces or has not yet been reported, try to isolate the problem before opening an issue, if possible. Also provide the steps taken to reproduce the bug.

Pull requests

We welcome bug fixes, improvements, and new features. Before embarking on making significant changes, please open an issue and ask first so that you do not risk duplicating efforts or spending time working on something that may be out of scope. For minor items, just open a pull request.

[Fork the project](#), clone your fork, and add the upstream to your remote:

```
$ git clone git@github.com:<your-username>/go-zfs.git
$ cd go-zfs
$ git remote add upstream https://github.com/mistifyio/go-zfs.git
```

If you need to pull new changes committed upstream:

```
$ git checkout master
$ git fetch upstream
$ git merge upstream/master
```

Don' work directly on master as this makes it harder to merge later. Create a feature branch for your fix or new feature:

```
$ git checkout -b <feature-branch-name>
```

Please try to commit your changes in logical chunks. Ideally, you should include the issue number in the commit message.

```
$ git commit -m "Issue #<issue-number> - <commit-message>"
```

Push your feature branch to your fork.

```
$ git push origin <feature-branch-name>
```

[Open a Pull Request](#) against the upstream master branch. Please give your pull request a clear title and description and note which issue(s) your pull request fixes.

- All Go code should be formatted using [gofmt](#).
- Every exported function should have [documentation](#) and corresponding [tests](#).

Important: By submitting a patch, you agree to allow the project owners to license your work under the [Apache 2.0 License](#).

Go Tools

For consistency and to catch minor issues for all of go code, please run the following:

- goimports
- go vet
- golint
- errcheck

Many editors can execute the above on save.

Guidelines based on http://azkaban.github.io/contributing.html