# Contributing

### Anyone is welcome to make BeEF better!

Thank you for wanting to contribute to BeEF. It's effort like yours that helps make BeEF such a great tool.

Following these guidelines shows that you respect the time of the developers developing this open source project and helps them help you. In response to this, they should return that respect in addressing your issue, assisting with changes, and helping you finalize your pull requests.

### We want any form of helpful contributions!

BeEF is an open source project and we love to receive contributions from the community! There are many ways to contribute, from writing tutorials or blog posts, improving or translating the documentation, answering questions on the project, submitting bug reports and feature requests or writing or reviewing code which can be merged into BeEF itself.

# Ground Rules

### Responsibilities

> \* When making an issue, ensure the issue template is filled out, failure to do so can and will result in a closed ticket and a delay in support.

> \* We now have a two-week of unresponsiveness period before closing a ticket, if this happens, just comment responding to the issue which will re-open the ticket. Ensure to make sure all information requested is provided.

> \* Ensure cross-platform compatibility for every change that's accepted. Mac and Linux are currently supported.

> \* Create issues for any major changes and enhancements that you wish to make. Discuss things transparently and get community feedback.

> \* Ensure language is as respectful and appropriate as possible.

> \* Keep merges as straightforward as possible, only address one issue per commit where possible.

> \* Be welcoming to newcomers and try to assist where possible, everyone needs help.

# Where to start

### Looking to make your first contribution

Unsure where to begin contributing to BeEF? You can start by looking through these issues:

\* Good First Issue - issues which should only require a few changes, and are good to start with.

\* Question - issues which are a question and need a response. A good way to learn more about BeEF is to try to solve a problem.

At this point, you're ready to make your changes! Feel free to ask for help; everyone is a beginner at first.

If a maintainer asks you to "rebase" your PR, they're saying that code has changed, and that you need to update your branch so it's easier to merge.

### Ruby best practise

Do read through: https://rubystyle.guide

Try and follow through with the practices throughout, even going through it once will help keep the codebase consistent.

Use Rubocop to help ensure that the changes adhere to current standards, we are currently catching up old codebase to match.

Just run the following in the /beef directory.

> rubocop

# Getting started

### How to submit a contribution.

1. Create your own fork of the code

2. Checkout the master branch

> git checkout master

3. Create a new branch for your feature

> git checkout -b my-cool-new-feature

4. Add your new files

> git add modules/my-cool-new-module

5. Modify or write a test case/s in Rspec for your changes

6. Commit your changes with a relevant message

> git commit

7. Push your changes to GitHub

> git push origin my-cool-new-feature

8. Run all tests again to make sure they all pass

9. Edit existing wiki page / add a new one explaining the new features, including:

- sample usage (command snippets, steps and/or screenshots)

- internal working (code snippets & explanation)

10. Now browse to the following URL and create your pull request from your fork to beef master

- Fill out the Pull Request Template

- https://github.com/beefproject/beef/pulls

# How to report a bug

If you find a security vulnerability, do NOT open an issue. Email security@beefproject.com instead.

When the security team receives a security bug email, they will assign it to a primary handler.

This person will coordinate the fix and release process, involving the following steps:

\* Confirm the problem and find the affected versions.

\* Audit code to find any potential similar problems.

\* Prepare fixes