# Contributing

## How to contribute

Contributing to the MSTG can be done in many different ways:

- \*\*File [issues](https://github.com/OWASP/owasp-mstg/issues "MSTG Issues")\*\* for missing content or errors. Explain what you think is missing and give a suggestion as to where it could be added.

- \*\*Create a [Pull Request (PR)](https://github.com/OWASP/owasp-mstg/pulls "Create a pull request")\*\*. This is a direct contribution to the guide and may be merged after review. Be sure to follow our [style guide](https://github.com/OWASP/owasp-mstg/blob/master/style\_guide.md "MSTG Style Guide") when writing content. You should ideally [create an issue](https://github.com/OWASP/owasp-mstg/issues "MSTG Issues") for any PR you would like to submit, as we can first review the merit of the PR and avoid any unnecessary work. This is of course not needed for small modifications such as correcting typos.

- \*\*Review pull requests for style, spelling and grammar\*\*. If you are a fluent speaker in any of the different languages that the MSTG is available in, feel free to give feedback on any of the submitted PRs.

- \*\*Review pull requests for technical content\*\*. Do you have tons of experience with the technical subject described in the PR? Chip in with constructive feedback.

- \*\*Proofread and fix errors\*\*. If you are studying the MSTG, write down any error, no matter how small, and submit them in an issue or fix them yourself through a PR.

- \*\*Contribute to the MSTG Apps\*\*. Extend and maintain our existing mobile applications, which are the [Crackmes](https://github.com/OWASP/owasp-mstg/tree/master/Crackmes "MSTG Crackmes") and the applications on our [Hacking playground](https://github.com/OWASP/MSTG-Hacking-Playground "MSTG Hacking Playground").

- \*\*Contribute to the Crackme Write-ups\*\*. If you create a write-up on how to crack the Crackmes or the Hacking Playground, we can gladly include a link to it: simply submit a PR adding your link to the [Crackmes README file](https://github.com/OWASP/owasp-mstg/tree/master/Crackmes).

## Making sure your PR is accepted

In order to increase the chances of your PR being accepted, please make sure that:

- Your submission is compliant with our [style guide](https://github.com/OWASP/owasp-mstg/blob/master/style\_guide.md "MSTG Style Guide").

- Your code snippets are well-tested and provide comments for key code lines.

- Your test cases explain issues on open-source or specifically designed vulnerable applications. Do not show vulnerabilities or bad coding practices on commercial applications. Example applications you can use are [OWASP iGoat](https://www.owasp.org/index.php/OWASP\_iGoat\_Tool\_Project "OWASP iGoat"), applications from the [MSTG Hacking playground](https://github.com/OWASP/MSTG-Hacking-Playground "MSTG Hacking Playground") or any of the [MSTG Crackmes](https://github.com/OWASP/owasp-mstg/tree/master/Crackmes "MSTG Crackmes").

After your PR has been submitted, we will review it as quickly as possible. This typically only takes a few days, but can vary depending on the size of the PR. Small PRs require only one reviewer, while large PRs may require multiple reviewers. We will always try to give initial feedback on your PR within 14 days. If you think we have forgotten about your PR, feel free to give us a nudge after these 7 days have passed.

## How to set up your contributor environment

1. Create a GitHub account. Multiple different GitHub subscription plans are available, but you only need a free one. Follow [these steps](https://help.github.com/en/articles/signing-up-for-a-new-github-account "Signing up for a new GitHub account") to set up your account.

2. Fork the repository. Creating a fork means creating a copy of the repository on your own account, which you can modify without any impact on this repository. GitHub has an [article that describes all the needed steps](https://help.github.com/en/articles/fork-a-repo "Fork a repo").

3. Clone your own repository to your machine so that you can make modifications. If you followed the GitHub tutorial from step 2, you have already done this.

4. Go to the newly cloned directory "owasp-mstg" and add the remote upstream repository:

```bash

$ git remote -v

origin git@github.com:<your Github handle>/owasp-mstg.git (fetch)

origin git@github.com:<your Github handle>/owasp-mstg.git (push)

$ git remote add upstream git@github.com:OWASP/owasp-mstg.git

$ git remote -v

origin git@github.com:<your Github handle>/owasp-mstg.git (fetch)

origin git@github.com:<your Github handle>/owasp-mstg.git (push)

upstream git@github.com:OWASP/owasp-mstg.git (fetch)

upstream git@github.com:OWASP/owasp-mstg.git (push)

```

See also the GitHub documentation on "[Configuring a remote for a fork](https://help.github.com/en/articles/configuring-a-remote-for-a-fork "Configuring a remote for a fork")".

5. Choose what to work on, based on any of the outstanding [issues](https://github.com/OWASP/owasp-mstg/issues "MSTG Issues").

6. Create a branch so that you can cleanly work on the chosen issue: `git checkout -b FixingIssue66`

7. Open your favorite editor and start making modifications. We recommend using the free [Visual Studio Code editor](https://code.visualstudio.com "Visual Studio Code") as it can make use of the code linting that is part of the repository through the [MarkdownLint plugin](https://github.com/DavidAnson/vscode-markdownlint#install "MarkdownLint plugin"). The code linter can help you when you make mistakes against our [style guide](https://github.com/OWASP/owasp-mstg/blob/master/style\_guide.md "MSTG Style Guide"), but be sure to read the style guide yourself, as the code linter will only detect a part of it.

8. After your modifications are done, push them to your forked repository. This can be done by executing the command `git add MYFILE` for every file you have modified, followed by `git commit -m 'Your Commit Message'` to commit the modifications and `git push` to push your modifications to GitHub.

9. Create a Pull Request (PR) by going to your fork, <https://github.com/Your\_Github\_Handle/owasp-mstg> and click on the "New Pull Request" button. The target branch should typically be the Master branch. When submitting a PR, be sure to follow the checklist that is provided in the PR template. The checklist itself will be filled out by the reviewer.

10. Your PR will be reviewed and comments may be given. In order to process a comment, simply make modifications to the same branch as before and push them to your repository. GitHub will automatically detect these changes and add them to your existing PR.

11. When starting on a new PR in the future, make sure to always keep your local repo up to date:

```bash

$ git fetch upstream

$ git merge upstream/master

```

See also the following article for further explanation on "[How to Keep a Downstream git Repository Current with Upstream Repository Changes](https://medium.com/sweetmeat/how-to-keep-a-downstream-git-repository-current-with-upstream-repository-changes-10b76fad6d97 "How to Keep a Downstream git Repository Current with Upstream Repository Changes")".

If at any time you want to work on a different issue, you can simply switch to a different branch, as explained in step 5.

> Tip: Don't try to work on too many issues at once though, as it will be a lot more difficult to merge branches the longer they are open.

## Translating

Our current goal is to publish one minor release every 6 months. Next, we will often create patch updates in order to provide intermediary updates in PDF and DocX format. Releases that have been tagged can then be translated into preferred languages.

> Note we use semantic versioning: major.minor.patch

## What not to do

Although we greatly appreciate any and all contributions to the project, there are a few things that you should take into consideration:

- The MSTG should not be used as a platform for advertisement of commercial tools, companies or individuals. Write-ups should be written with free and open-source tools in mind and commercial tools are typically not accepted, unless as a reference in the security tools section.

- Unnecessary self-promotion of tools or blog posts is frowned upon. If you have a relation with on of the URLs or tools you are referencing, please state so in the PR so that we can verify that the reference is in line with the rest of the guide.

Please be sure to take a careful look at our [Code of Conduct](https://github.com/OWASP/owasp-mstg/blob/master/CODE\_OF\_CONDUCT.md "Code of Conduct") for all the details.