# Submitting New CVEs Using GitHub Desktop

*Author: Madison Oliver; July 2019*

The purpose of this document is to describe the GitHub submission process using GitHub Desktop, free GUI-based software. This can also be replicated with other GUIs → <https://git-scm.com/downloads/guis>. The following steps assume that your git environment is already set up (if it isn’t, please review the Setup document).

1. While working in your repository, create a new branch and check it out
	1. Create a branch to work in named "cna/[ORG]/[SOME IDENTIFIER]", where "SOME IDENTIFIER" could be:
		1. the CVE ID if it is a single CVE
		2. The organization’s publication tracking number if it is multiple CVEs from one single vulnerability publication
		3. A description of its purpose if it is from multiple publications
		4. The date of the submission in YYYYMMDD format
		5. **
		6. 
2. Move your completed .json files to the appropriate directory, e.g. CVE-2019-1234.json goes into YOUR-REPOSITORY/2019/1xxx/
	1. This can be accomplished on the command line by using cp or mv or in your file explorer with copy/paste or drag and drop
	2. You will have to replace the file, as one already exists with filler text
3. Add these files to the staging environment and commit the files to your repository. Checking the box next to the new file will add it to the staging environment.
	1. To commit, create your commit message in the empty field above description, add a description if needed, then click the blue “Commit” button
	2. 
	3. If you have already set up your git environment to sign commits, you should be prompted for your PGP passphrase to sign the commit. If you haven’t done this yet, please review the Setup document for help.
4. Push the new branch to your organization’s remote repository on GitHub by clicking “Publish branch”.
	1. 
5. On GitHub (<https://github.com>), you should be able to initiate a pull request to MITRE’s repository
	1. There should be a green box that says "Compare and pull request" that you can click to generate the pull request
		1. 
			1. If that box is not there, just click "New pull request". The box is generated by GitHub acknowledging differences between MITRE's CVE list and your forked copy. Sometimes it isn't immediate recognize it depending on how quickly you do this, so if that box is not there it doesn't necessarily mean the changes aren't there.
	2. Initial the pull request, put an appropriate message in there (it can be the same as the commit message) and submit it to MITRE
		1. They will come back with any requested changes if needed. To make these changes, update the files on your host machine in the same repo and branch that you initially put it in, commit the new changes, and then push them to Github. They'll automatically show up as part of this pull request as a new commit.

# Updating

If you want to update an **existing** CVE (that is, one that exists in MITRE’s Github already and is completed), you should **first** pull the latest updates from MITRE’s repo (<https://github.com/CVEProject/cvelist>) to avoid any merge conflicts (<https://www.atlassian.com/git/tutorials/using-branches/merge-conflicts>), then update the file and submit it using the aforementioned steps.

1. Add MITRE’s CVE list as a remote repo that you can pull from, named “mitre” below
	1. **$ git remote add mitre** [**https://github.com/CVEProject/cvelist.git**](https://github.com/CVEProject/cvelist.git)
2. Pull the MITRE branch that you want to receive updates from, almost always the master branch
	1. **$ git pull mitre master**
3. Then finally push these changes to your organization’s Github
	1. **$ git push**