

Help Topic: Enable Interactive Modes

Spring 2020

Michael Benjamin, mikerb@mit.edu
Department of Mechanical Engineering
MIT, Cambridge MA 02139

Don't Shoot Yourself in the Foot - Enable Interactive Modes

A couple notes of caution. The **rm**, **mv**, and **cp** commands can be brutally powerful. They are neither merciful nor merciless. If used with a bit care they are as reliable as an old friend. It turns out, you can easily tweak their default behavior to make them a bit more merciful to handle common mistakes. Having used these tools for about 20 years, I still use them in this way. If you're a *new* user, I highly recommend this practice.

Each of these three tools gives you the power to either remove or overwrite an existing file. By default, they will just do as told without prompting the user for further confirmation of intention. For example, suppose you want to rename `file1` to `file2`, but you forgot that you already have a file by the name of `file2`:

```
$ ls
file1 file2
$ mv file1 file2
$ ls
file2                (Yikes!)
```

The **mv**, **rm**, and **cp** commands all take an option argument **-i**, which prompts the user when there is a possibility of overwriting or removing a file. That's wonderful, but if we're trying to prevent silly mistakes, what about the silly mistake of forgetting to use the **-i** command line option?

None of these three commands have any way to modify them to use **-i** by default. However, the same effect can be achieved by creating an alias for them and putting the aliases in your shell configuration file.

Creating an alias for the **rm**, **mv**, and **cp** commands

Our goal is to make the **-i**, interactive mode, the default behavior used by **rm**, **mv**, and **cp**. To do this, we use an *alias*. If you haven't encountered aliases, they are an absolute must-have tool in your command line toolbox. If you're reading this page, the page on aliases shouldn't be too far off on your list of things to learn:

http://oceanai.mit.edu/ivpman/help/cmdline_aliases

You can create an alias for the **rm** command right now by typing the following on the command line. Similar syntax is used for the **mv** and the **cp** commands.

```
$ alias rm 'rm -i'      (if using tcsh)
$ alias rm='rm -i'      (if using bash)
```

However, this alias will only hold as long as you are using this shell (Terminal) session. If you open a new window, or log in and log out, **rm** will revert back to its old self and not prompt you for confirmation. To ensure **rm** always prompts for recommendation, you'll need to put that alias in your shell configuration file.

Permanently adding **rm**, **mv**, and **cp** aliases to your shell

Customizing your shell environment may also be a topic you haven't hit yet, but it is super important and comes with the territory of using aliases. More can be found here:

http://oceanai.mit.edu/ivpman/help/cmdline_shell_environment If you're a tcsh user, add the following lines in your shell configuration file which is typically, `/.cshrc`:

```
alias rm 'rm -i'
alias cp 'cp -i'
alias mv 'mv -i'
```

If you're a bash user, add the following lines in your shell configuration file which is typically, `/.bashrc`:

```
alias rm='rm -i'
alias cp='cp -i'
alias mv='mv -i'
```

Finally, sometimes the default behavior (no prompting for confirmation) really is what you want. The prompting may be annoying if operating on several files. If the operation is part of a script, it can hold up the script. In these cases, the **-f**, or *force* option, can be used to override the interactive option.