Linux Learning Game

By: Samuel Bechara

About

This was made to be a interactive way to learn how to navigate Linux filesystems. I am probably walking around the room right now weirding you out. Or maybe you are an online student. In either way, I hope you learn a little, and have a little fun.

The first rule of Linux: Everything is either a file or a folder

This is really one of the only things you need to know! A file is a thing (a picture, a word document, a video) and a folder is a place that files can live in. Everything is a file, and it has to live in a folder. You are always in a folder (the very base folder is `/`).

As an example, I am currently writing this document in my `/Users/sbechara/Documents/CSU/Teaching/MECH105/linuxGame` folder (that long description is called the path, notice that the lowest level is `/` showing that there is nothing lower than that). So to break it down, the `linuxGame` folder is in the `MECH105` folder which is in the `Teaching` folder which is in...you get the idea. Do you?

If you do, congratulations! You are halfway to being a linux expert! But the other half is kind of hard, but still! Halfway there!

man - talk to the man!

What I mean is that on linux systems, if you forget what something does, you can read the manual! I'll admit that it is pretty archaic and hard to understand, but the more you read them the more sense it makes.

Basic Commands

I have listed some basic commands below. For each of them, try reading the man page by typing in `man <command>`.

`pwd` - print working directory (tells you where you currently are)
`cd` - change directory (allows you to move around)
`touch` - create a file
`ls` - list files and subdirectories in the current working directory
`cat` - print out the contents of a file
`mkdir` - make a directory
`rm` - remove a file
`cat` - print a file to the terminal window (it actually does more than that but for our purposes it prints)

Although there are many more commands you can learn, these are the basics and with them, you can start exploring Linux filesystems! Learn all about the previous commands by typing man and reading. You press the letter `q` on your keyboard to quit the man program.

Using Command Basics - Specifying Inputs

Some commands work by themselves but most of the time you want to do *something* with your commands. For example, if we look at what the command `cd` does, we can see that it changes your directory (you can think of that as where you currently are in the filesystem). We need a way to specify *where* you want to go! All Linux filesystems have a base folder signified by `/`. Everything (and I mean everything) is in that folder. So, lets say we wanted to move to the folder `/home/sbechara/Documents` (that is the folder's "path"). In order to do so we would type the path after `cd` like this: `cd /home/sbechara/Documents`.

A lot of the time we are going to be specifying either paths or files as inputs to our functions.

Using Command Basics - Specifying Options

It is sometimes useful to change up how a command works. For example, go ahead and go home by typing in `cd ~`. It is important to remember that `~` is a shortcut for your home folder (There is no place like `~`, there is no place like `~`). Go ahead and create a folder called `hello` by typing in the command `mkdir hello`. You can see your new folder by typing in `ls`.

If you try and delete the folder named `hello/` without specifying a special option (coming soon), you will get the following error message: `rm: hello/: is a directory`. You can see so for yourself by typing in `rm hello/` (the `/` just signifies that it is a folder and is actually unnecessary but is good practice). Basically, Linux is trying to protect you by telling you that you are trying to remove a directory which is dangerous! Most of the time you want to delete files. Not only that, but think about how much worse it can be if you delete a directory! You could end up deleting a lot of files if that directory has subdirectories, etc. So we have to specify, that we really understand it is a directory by issuing the "directory" option. Go ahead and read the `man rm` page to see if you can find the option that will work.

Hopefully you actually tried, you should see we need the `-d` option. So to remove the aforementioned directory we actually need to type: `rm -d hello/`. Options are almost always specified by a `-` symbol and a letter.

Ok, the manual kind of sucks...

Now that you have read a few `man` pages, I think you will agree that the manual is kind of hard to read and understand. A couple of nerds (said endearingly) have created a program called `tldr` that makes man pages way easier to read.

If you are using Ubuntu, you can install `tldr` by typing in the following command: `sudo apt install tldr`. To break down this command you are typing in:

- `sudo` - that is short for super-user do and is saying that you want to run the next command as an administrator. You need administrative privelages to install software.
- `apt` - that is short for "aptitude" which is the name of the package manager for Ubuntu. It is a program the manages all the software on your computer. You can install and remove programs with `apt`
- `install` - that is an input for the `apt` program. It is telling `apt` that you want it to search the internet for a program (the next input) and download it.
- `tldr` - this is the input to `install` and specifies which program we want to install!

If you are on a Mac, make sure you have homebrew installed. Then you can type the following command: `brew install tldr`

Once you have it installed, you can check out the `tldr` for any of the commands above by typing in `tldr <command>`

It is time to go on a journey!

I think it is best to think of Linux filesystems as a bunch of branching trails. Think of this as a journey. You are going to start in the folder called `linuxGameHome`. Let's see what we have in our current folder and where we can go!

To get started:

1. First, go home by typing in `cd ~`. Do an `ls` to see what is in your home folder. Make a mental note of what is there.
2. Now, we need to "clone" the repository that I created that has all the files you will need in them. You can do so by typing in `git clone https://github.com/sbechara/linuxGame.git`
3. Enter the `ls` command again. Magic! This is one of the reasons linux is cool because you can do stuff like this...

Now that you have the basics, it is time for your journey! If you want to see this file printed out to the terminal window you can type `cat ReadMe.md` (Recall, the command is `cat` and the input is `ReadMe.md`). Now, your journey is up to you! You can go through the door if you want, but I don't think there is anything there... Probably should start by going into the field. Don't forget to type `ls` once you get there to see what is available to you and you can read files by typing `cat <filename>`!