Linux

General linux Information for someone that is new to linux.

- [Command line crash course](#)
- [How do I use "screen" to keep applications running after I close my SSH session?](#)

Command line crash course

A quick introduction to the linux command line and frequently used commands.

Paths

There are full paths and partial paths in linux. These behave similar to windows and other OSs

- Full paths begin with `/` in linux and specify the exact location of a file or folder (think of any paths that being with C:/ in windows)
- Partial paths are relative to the currently directory they do not begin with any special symbols. (ie `cd somefolder/somefile` would tell linux to append somefolder/somefile to your current path and go there)
- You can also specify current directory with a dot. (ie `cd ./somefolder/somefile` is the same as the previous bullet point)
- Previous directory is specified with two dots (ie `cd ..` will take you up a directory)
- You can also specify your home directory with `~`. Your home directory is the one that the terminal puts you in when you first login (usually /home/username)

Basic Common Commands

Here are some basic commands in linux that most people will want to use

`cd <directory>`

Change directories.

- `cd .` does not do anything
- `cd ..` takes you up a directory
- `cd ~` takes you back to your home directory

```
amdhome@biostats:~$ cd /srv/shiny-server/spins-abcd/
amdhome@biostats:/srv/shiny-server/spins-abcd$ cd .
amdhome@biostats:/srv/shiny-server/spins-abcd$ cd ..
amdhome@biostats:/srv/shiny-server$ cd ~
amdhome@biostats:~$
```

`pwd`

Print working directory: prints the current folder you are in

```
amdhome@biostats:~$ pwd
/home/amdhome
```

`ls`

List files and folders in the current directory You can also specify a full or partial path to print that directories files/folders instead You can append `-al` for more information on the files

```
amdhome@biostats:/srv/shiny-server$ ls
index.html  sample-apps  spins-abcd

amdhome@biostats:/srv/shiny-server$ ls -al
total 12
drwxr-xr-x 3 root root 4096 Jun 13 13:37 .
drwxr-xr-x 3 root root 4096 Jun 13 12:41 ..
lrwxrwxrwx 1 root root   38 Jun 13 12:41 index.html -> /opt/shiny-server/samples/welcome.html
lrwxrwxrwx 1 root root   37 Jun 13 12:41 sample-apps -> /opt/shiny-server/samples/sample-apps
drwxr-xr-x 2 root root 4096 Jun 13 14:07 spins-abcd
```

`mv <source> <destination>`

Move file/folder. Both You also use this to rename things

```
amdhome@biostats:~$ ls
app.R

amdhome@biostats:~$ mv app.R app2.R
amdhome@biostats:~$ ls
```

```
  app2.R

  amdhome@biostats:~$ mv app2.R /tmp/
  amdhome@biostats:~$ ls /tmp
  app2.R
```

`cp <source> <destination>`

Copies files. You can specify `-r` if you want to copy directories

`rm <target>`

Removes files. You can append `-r` to remove directories

`mkdir <target>`

Creates a folder

`sudo`

Sudo is a special command that lets you run commands with elevated (root) permissions. All you have to do is put `sudo` at the beginning of a command. An administrator must have already granted you sudo rights in order to use this command

Exception

The only exception to this is `cd`. This is because for most commands that you run, you are telling your shell (command line/terminal) to run a program. `sudo` is a program that will run another program with elevated permissions. The `cd` command is technically not a program, but a command to tell your shell to change state. Because `cd` is not a program it can not be sudo-ed.

If you need to `cd` to a directory that requires elevated permissions, you can just become root by using the command `sudo -i`:

```
  amdhome@biostats:~$ sudo -i
  root@biostats:~$
```

Do be careful of any action taken while as root or when running with sudo as many sanity checks are bypassed when using linux as root.

How do I use "screen" to keep applications running after I close my SSH session?

Normally if you leave an SSH session, everything that is running will get killed. If you want to have a process continue to run you will need to use a program like screen. This will create a persistent session that stays running even after you log out of your ssh session

Commands

Launching a Screen Session

To launch a screen session run:

```
screen -S <name>
```

```
# Example
screen -S project1
```

The name can be whatever you want as long as there is no spaces. You will use this to identify and reconnect to your session at a later time.

After hitting enter it will look like a normal command line, go ahead and run whatever long process you were going to here

Detaching From a Screen Session

This is like the minimize button in Windows, your screen session is still running in the background.

After running your program, you will probably want to detach from the screen session to do other things. The easiest way is to just close your ssh window.

To detach the session press `ctrl` + `a` , let go of the keys and then press `d` . This will send you back to your normal ssh session.

Do not type `exit` or `logout` in a screen session until you are ready to end the screen session. `exit` and `logout` will cause screen to terminate and take you back to the normal ssh session.

Exiting Screen Sessions

As stated above, to terminate the screen session and all commands running within it type `exit` or `logout` . Screen will exit and kill any processes running in it.

Listing Your Active Screen Sessions

You may want to see what sessions you have active. To do this run:

```
screen -ls
```

In the example below I have 2 screen sessions. One named **test**, and one named **project1**:

```
[amdhome@mercury ~]$ screen -ls
There is a screen on:
        9955.test       (Detached)
        9971.project1   (Detached)
1 Socket in /var/run/screen/S-amdhome.
```

Reattaching to a Screen Session

To reattach to your screen session run:

```
screen -r <name>
```

```
# Example

screen -r project1
```

This should take you back to your screen session

Scrolling in a screen window

Sometimes you need to scroll up in a screen window, but screen won't let you by default.

To scroll press `ctrl` + `a` let go of the keys and then press `esc` . You should now be able to scroll with your arrow keys