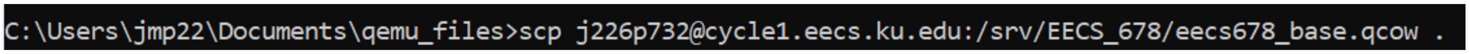
Joseph Pennington

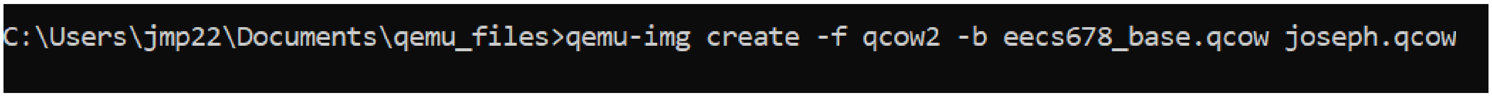
2912079

EECS 678 Lab02



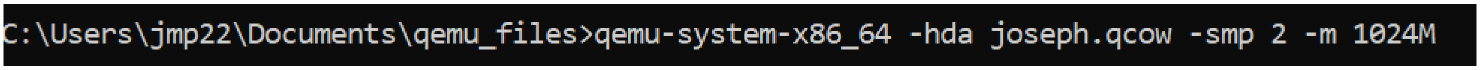
```
C:\Users\jmp22\Documents\qemu_files>scp j226p732@cycle1.eecs.ku.edu:/srv/EECS_678/eecs678_base.qcow .
```

1. After downloading and installing QEMU, the command copied the base image from the cycle servers onto my machine



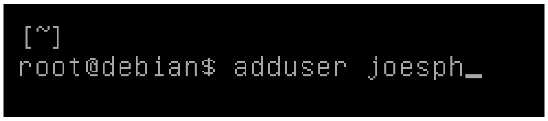
```
C:\Users\jmp22\Documents\qemu_files>qemu-img create -f qcow2 -b eecs678_base.qcow joseph.qcow
```

2. Once the image was copied, this command makes a differential image off the local copy of the cycle server base image. The new image is named “joseph.qcow”.

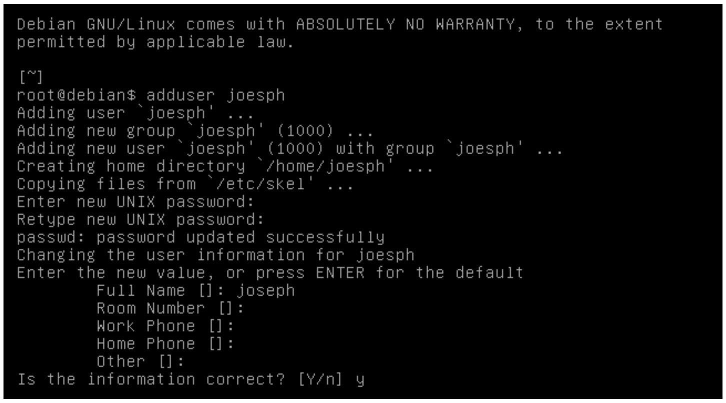


```
C:\Users\jmp22\Documents\qemu_files>qemu-system-x86_64 -hda joseph.qcow -smp 2 -m 1024M
```

3. This command boots up the copied image from the previous command. In my case, using the -redir flag caused an error. However, by removing it, it caused the GUI to open with 1024M of memory.

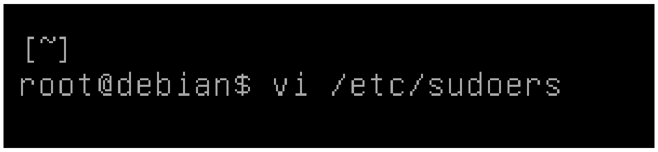


```
[~]
root@debian$ adduser joesph_
```

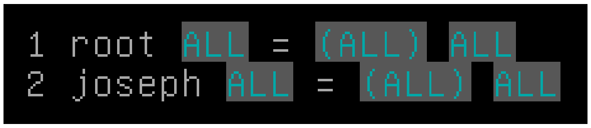


```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

[~]
root@debian$ adduser joesph
Adding user `joesph' ...
Adding new group `joesph' (1000) ...
Adding new user `joesph' (1000) with group `joesph' ...
Creating home directory `/home/joesph' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for joesph
Enter the new value, or press ENTER for the default
        Full Name []: joseph
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
```

4. Once the GUI was opened and loaded, I added myself as a user in addition to the root user.

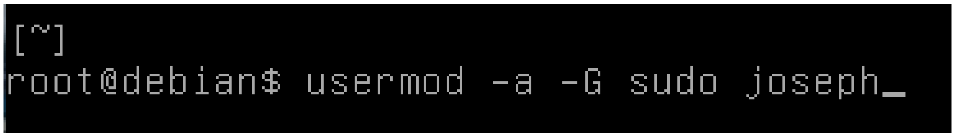


```
[~]
root@debian$ vi /etc/sudoers
```



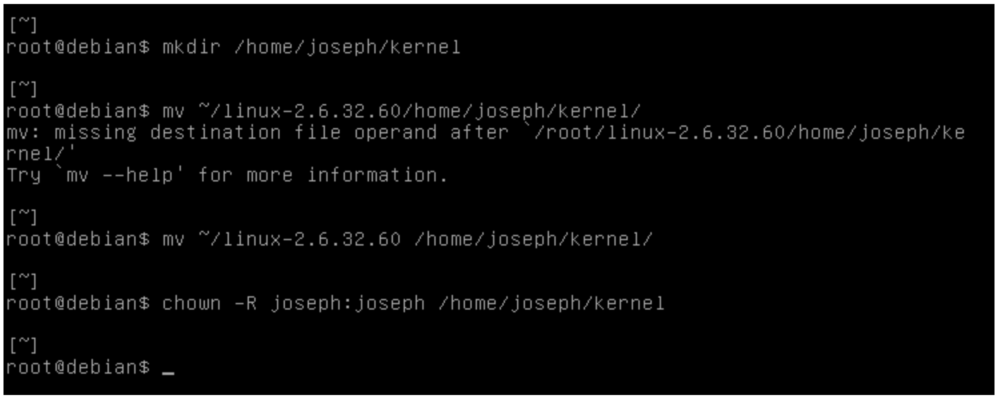
```
root ALL = (ALL) ALL
joseph ALL = (ALL) ALL
```

5. This command opens vi on the /etc/sudoers file. I then added the two lines to add the new user to the file.



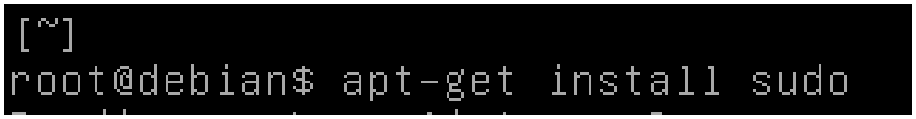
```
[~]
root@debian$ usermod -a -G sudo joseph_
```

6. Once the user was added to the file, this command adds the new user to the sudo group. This gives the new user system admin rights.



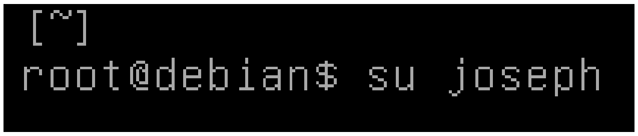
```
[~]
root@debian$ mkdir /home/joseph/kernel

[~]
root@debian$ mv ~/linux-2.6.32.60/home/joseph/kernel/
mv: missing destination file operand after `/root/linux-2.6.32.60/home/joseph/ke
rnel/'
Try `mv --help' for more information.

[~]
root@debian$ mv ~/linux-2.6.32.60 /home/joseph/kernel/

[~]
root@debian$ chown -R joseph:joseph /home/joseph/kernel

[~]
root@debian$ _
```

7. The next step was to move the kernel source and the config file to the user’s home directory. Then I changed the ownership to the user.



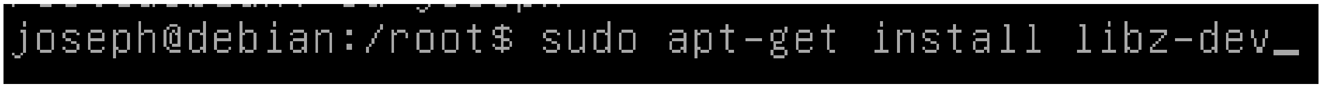
```
[~]
root@debian$ apt-get install sudo
```

8. This command installs sudo onto the kernel. This enables the new user to call commands with the rights from the previous command.



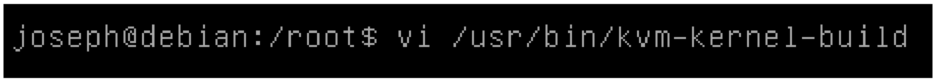
```
[~]
root@debian$ su joseph
```

9. After installing sudo, this command switches the root user account to the new account I have created.



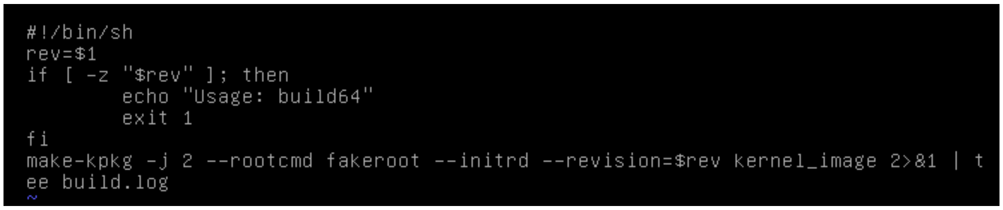
```
joseph@debian:/root$ sudo apt-get install libz-dev_
```

10. On the new user account, this command prevents an error that may happen during installation. This installs the libz package.



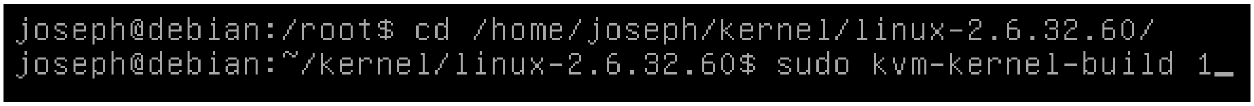
```
joseph@debian:/root$ vi /usr/bin/kvm-kernel-build
```

11. This command opens the kvm-kernel-build script using vi so I can add the necessary changes before building the kernel.

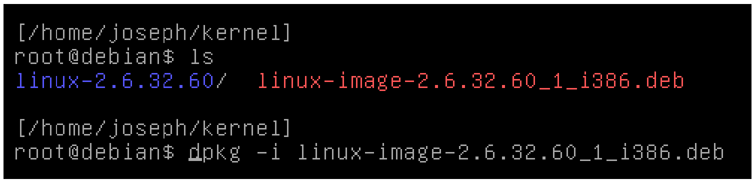


```
#!/bin/sh
rev=$1
if [ -z "$rev" ]; then
        echo "Usage: build64"
        exit 1
fi
make-kpkg -j 2 --rootcmd fakeroot --initrd --revision=$rev kernel_image 2>&1 | t
ee build.log
~
```

12. Once the script was opened, I added to -j flag with the value 2 to the make-kpkg command.

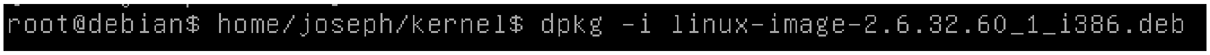


```
joseph@debian:/root$ cd /home/joseph/kernel/linux-2.6.32.60/
joseph@debian:~/kernel/linux-2.6.32.60$ sudo kvm-kernel-build 1_
```



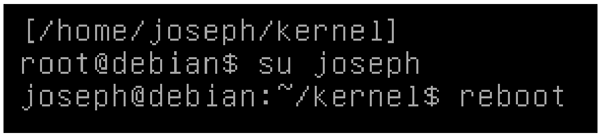
```
[/home/joseph/kernel]
root@debian$ ls
linux-2.6.32.60/  linux-image-2.6.32.60_1_i386.deb

[/home/joseph/kernel]
root@debian$ dpkg -i linux-image-2.6.32.60_1_i386.deb
```

13. Once the change to the script was saved, the first command changes the directory to the kernel source directory. The second command builds the kernel using the sudo rights as the new user. After this build finishes, a new .deb file should be created in the directory above the kernel source.

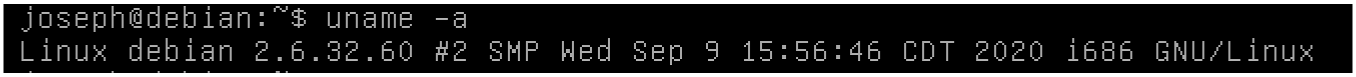


```
root@debian$ home/joseph/kernel$ dpkg -i linux-image-2.6.32.60_1_i386.deb
```

14. After the new .deb file was created, this command installs the kernel.



```
[/home/joseph/kernel]
root@debian$ su joseph
joseph@debian:~/kernel$ reboot
```



```
joseph@debian:~$ uname -a
Linux debian 2.6.32.60 #2 SMP Wed Sep 9 15:56:46 CDT 2020 i686 GNU/Linux
```

15. The last steps were to check if the KVM had booted the newly built kernel. First, the system was rebooted from the new account using the first command. Once the system reloaded, I selected the Linux Debian 2.6.32.60 option from the boot menu. Then, I logged into the new "joseph" account. Lastly, to double check that I was running the correct kernel version, I ran the second command.