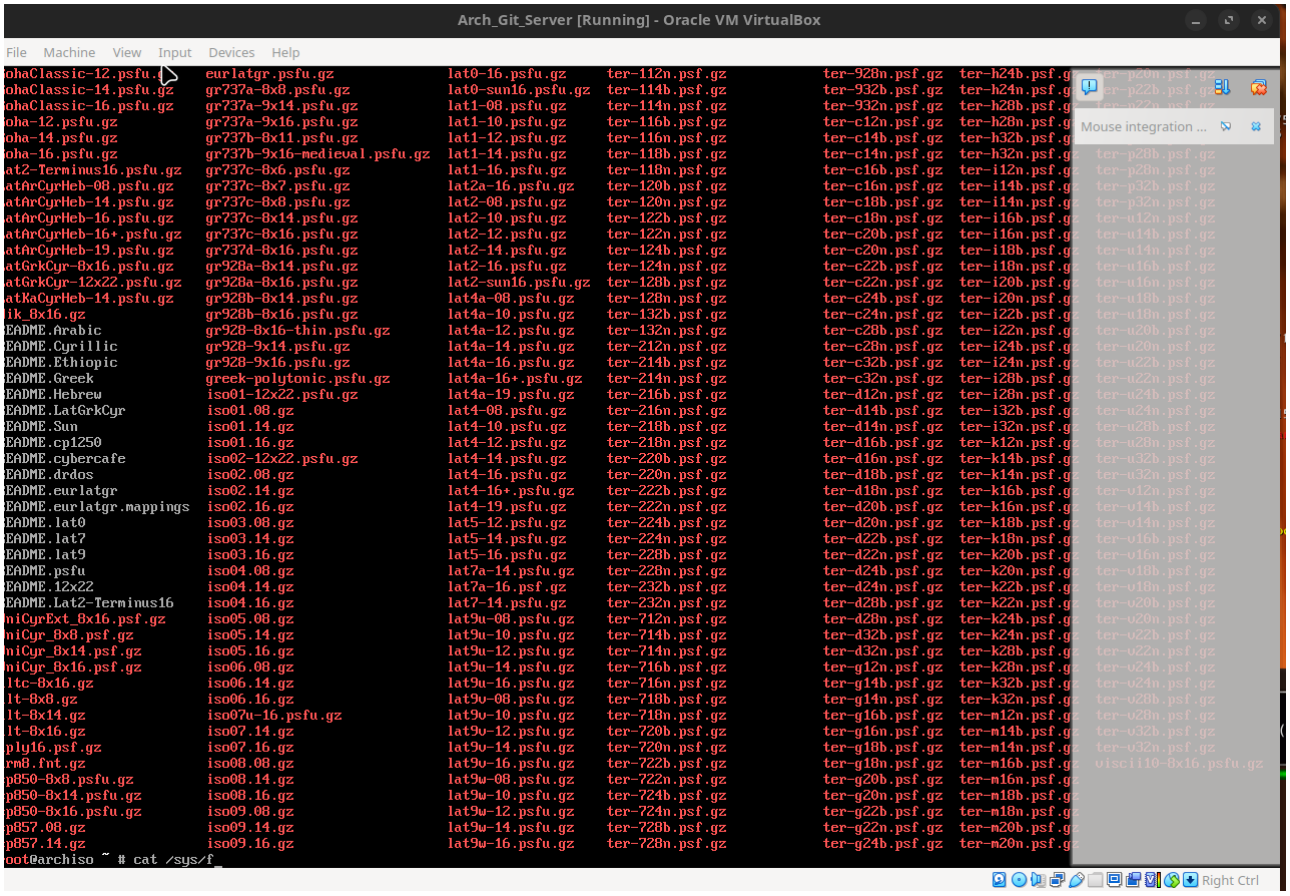
Arch Linux install/Git server setup

For my final project I set install/setup Arch Linux on a VM, I then set up a git server on that VM.

The projects parts will be listed out as steps some with pictures demonstrating what is happening in the step.

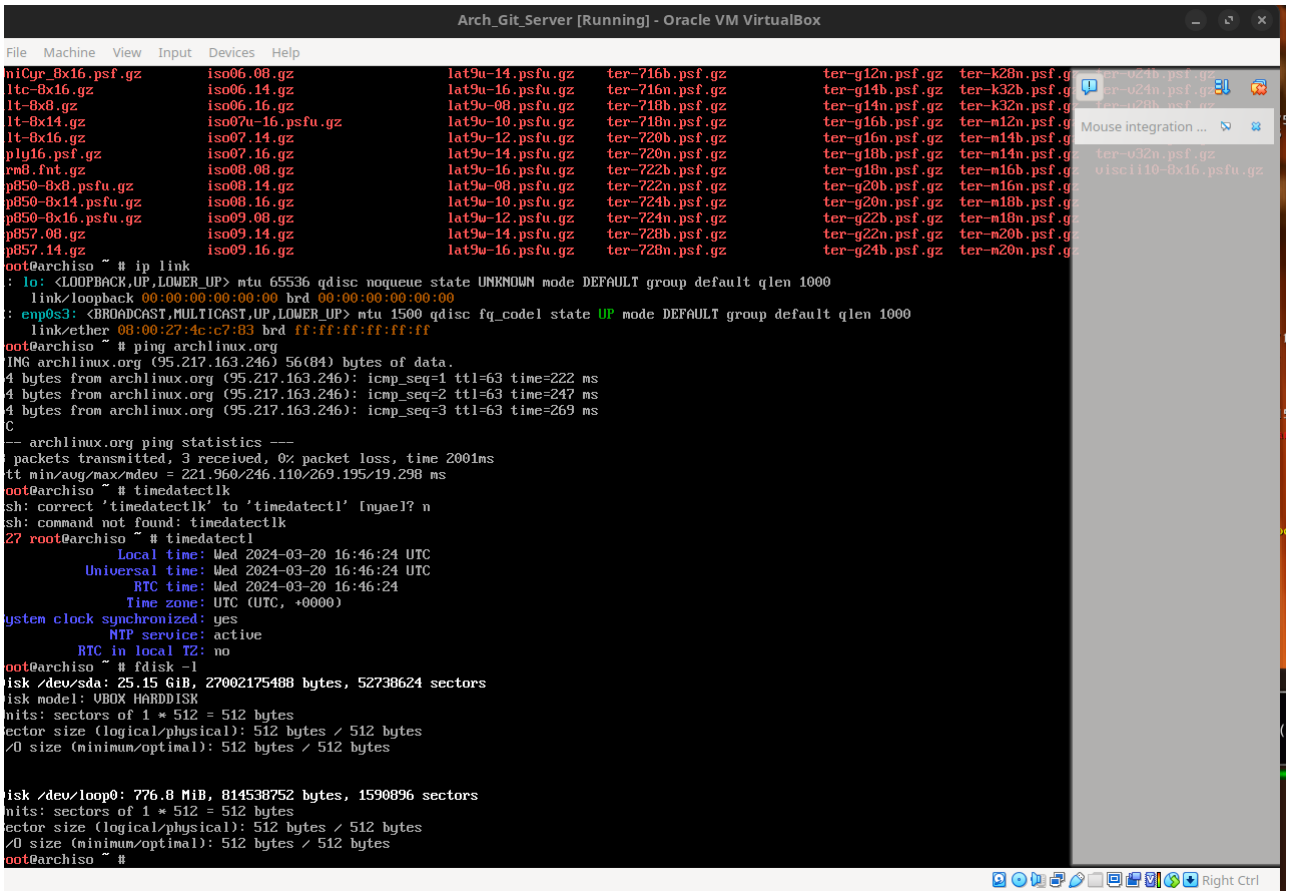
1. Install arch Linux ISO
2. Create VM and inject ISO in to the VM
3. Turn on VM
4. Load Arch live boot (From steps 4-21 I was referring to the Arch Linux installation guide https://wiki.archlinux.org/title/Installation_guide)
5. Configure language/font of the OS (pic 1)

Pic 1



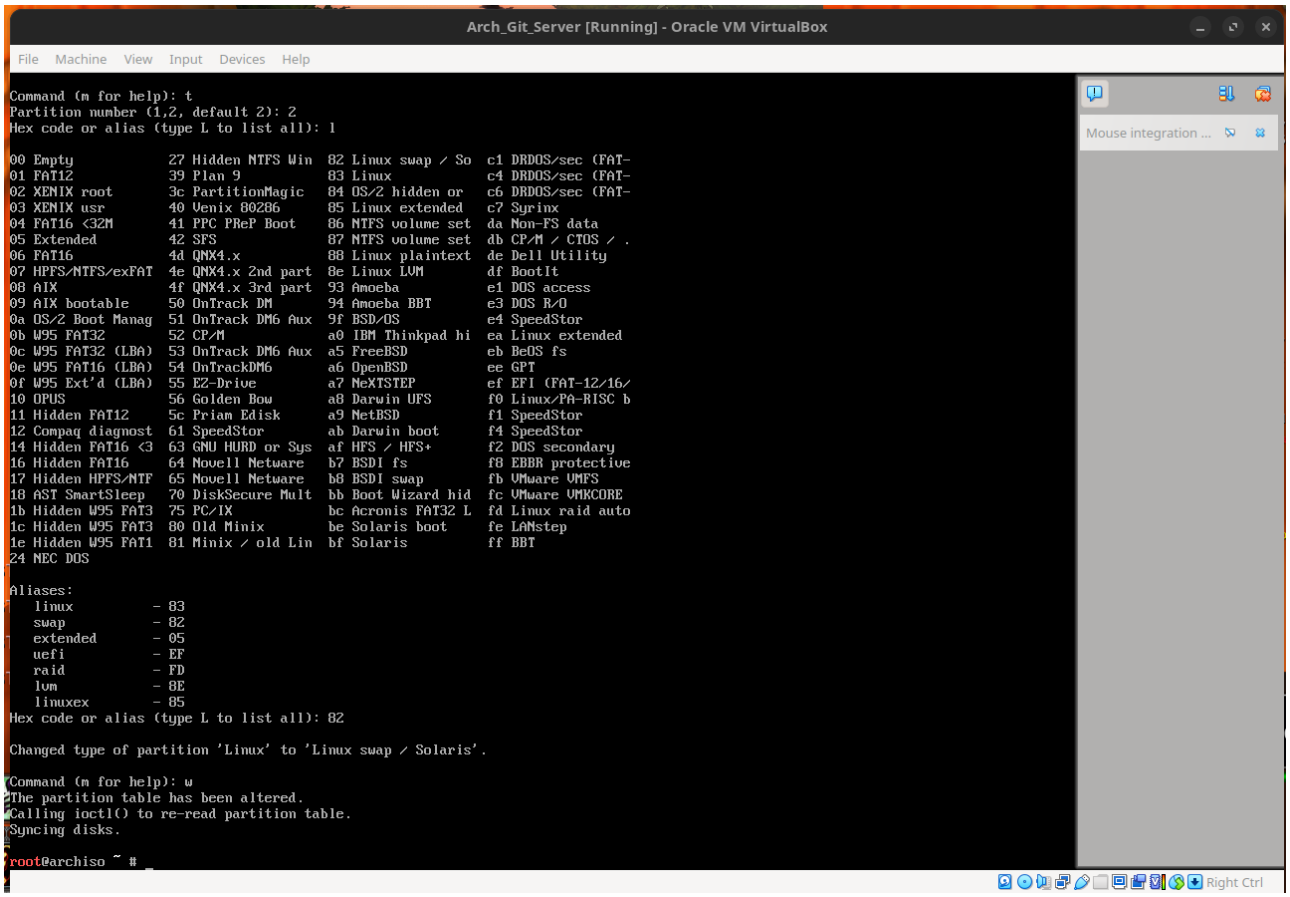
6. Connect to the internet by running this command “ip link” (pic 2)
7. List the “disks” on the machine by running this command “fdisk -l”(pic 2)

Pic 2



8. Select the disk to partition by running this command “fdisk /dev/sda”
9. Using fdisk partition the disk into 2 separate partitions one for swap and the other for /mnt
10. Using fdisk format the swap to a swap type partition then write the changes (pic 3)

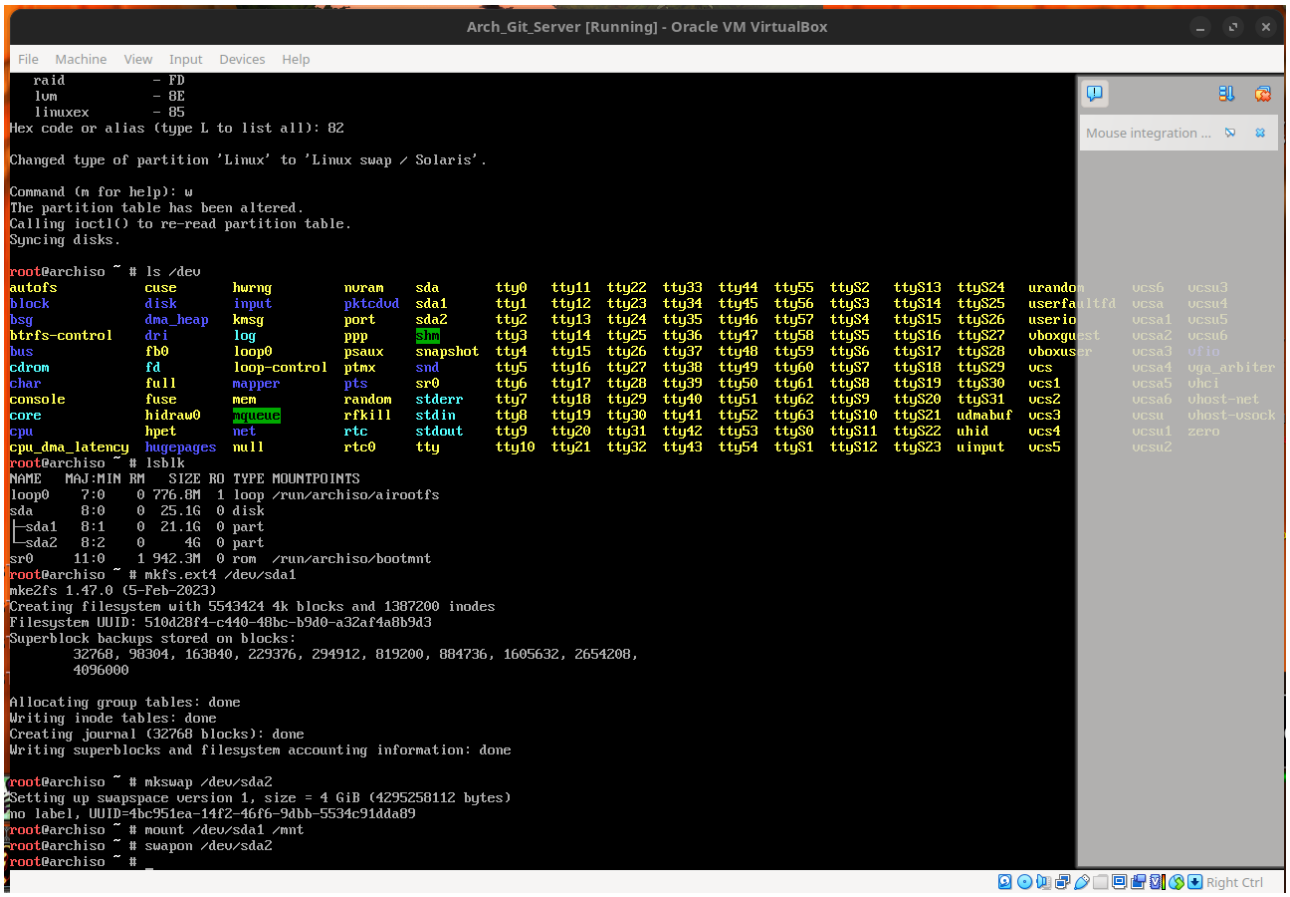
Pic 3




```
Command (m for help): t
Partition number (1,2, default 2): 2
Hex code or alias (type L to list all): l

00 Empty            27 Hidden NTFS Win  82 Linux swap / So  c1 DRDOS/sec (FAT-
01 FAT12            39 Plan 9           83 Linux            c4 DRDOS/sec (FAT-
02 XENIX root       3c PartitionMagic   84 OS/2 hidden or   c6 DRDOS/sec (FAT-
03 XENIX usr        40 Venix 80286      85 Linux extended   c7 Syrinx
04 FAT16 <32M       41 PPC PReP Boot    86 NTFS volume set  da Non-FS data
05 Extended         42 SFS              87 NTFS volume set  db CP/M / CTOS / .
06 FAT16            4d QNX4.x           88 Linux plaintext  de Dell Utility
07 HPFS/NTFS/exFAT  4e QNX4.x 2nd part  8e Linux LVM        df BootIt
08 AIX              4f QNX4.x 3rd part  93 Amoeba           e1 DOS access
09 AIX bootable     50 OnTrack DM       94 Amoeba BBT       e3 DOS R/O
0a OS/2 Boot Manag  51 OnTrack DM6 Aux  9f BSD/OS           e4 SpeedStor
0b W95 FAT32        52 CP/M             a0 IBM Thinkpad hi  ea Linux extended
0c W95 FAT32 (LBA)  53 OnTrack DM6 Aux  a5 FreeBSD          eb BeOS fs
0e W95 FAT16 (LBA)  54 OnTrackDM6       a6 OpenBSD          ee GPT
0f W95 Ext'd (LBA)  55 EZ-Drive         a7 NeXTSTEP         ef EFI (FAT-12/16/
10 OPUS             56 Golden Bow       a8 Darwin UFS       f0 Linux/PA-RISC b
11 Hidden FAT12     5c Priam Edisk      a9 NetBSD           f1 SpeedStor
12 Compaq diagnost  61 SpeedStor        ab Darwin boot      f4 SpeedStor
14 Hidden FAT16 <3  63 GNU HURD or Sys  af HFS / HFS+       f2 DOS secondary
16 Hidden FAT16     64 Novell Netware   b7 BSDI fs          f8 EBBR protective
17 Hidden HPFS/NTF  65 Novell Netware   b8 BSDI swap        fb VMware VMFS
18 AST SmartSleep   70 DiskSecure Mult  bb Boot Wizard hid  fc VMware VMKCORE
1b Hidden W95 FAT3  75 PC/IX            bc Acronis FAT32 L  fd Linux raid auto
1c Hidden W95 FAT3  80 Old Minix        be Solaris boot     fe LANstep
1e Hidden W95 FAT1  81 Minix / old Lin  bf Solaris          ff BBT
24 NEC DOS

Aliases:
   linux          - 83
   swap           - 82
   extended       - 05
   uefi           - EF
   raid           - FD
   lvm            - 8E
   linuxex        - 85
Hex code or alias (type L to list all): 82

Changed type of partition 'Linux' to 'Linux swap / Solaris'.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

root@archiso ~ # _
```

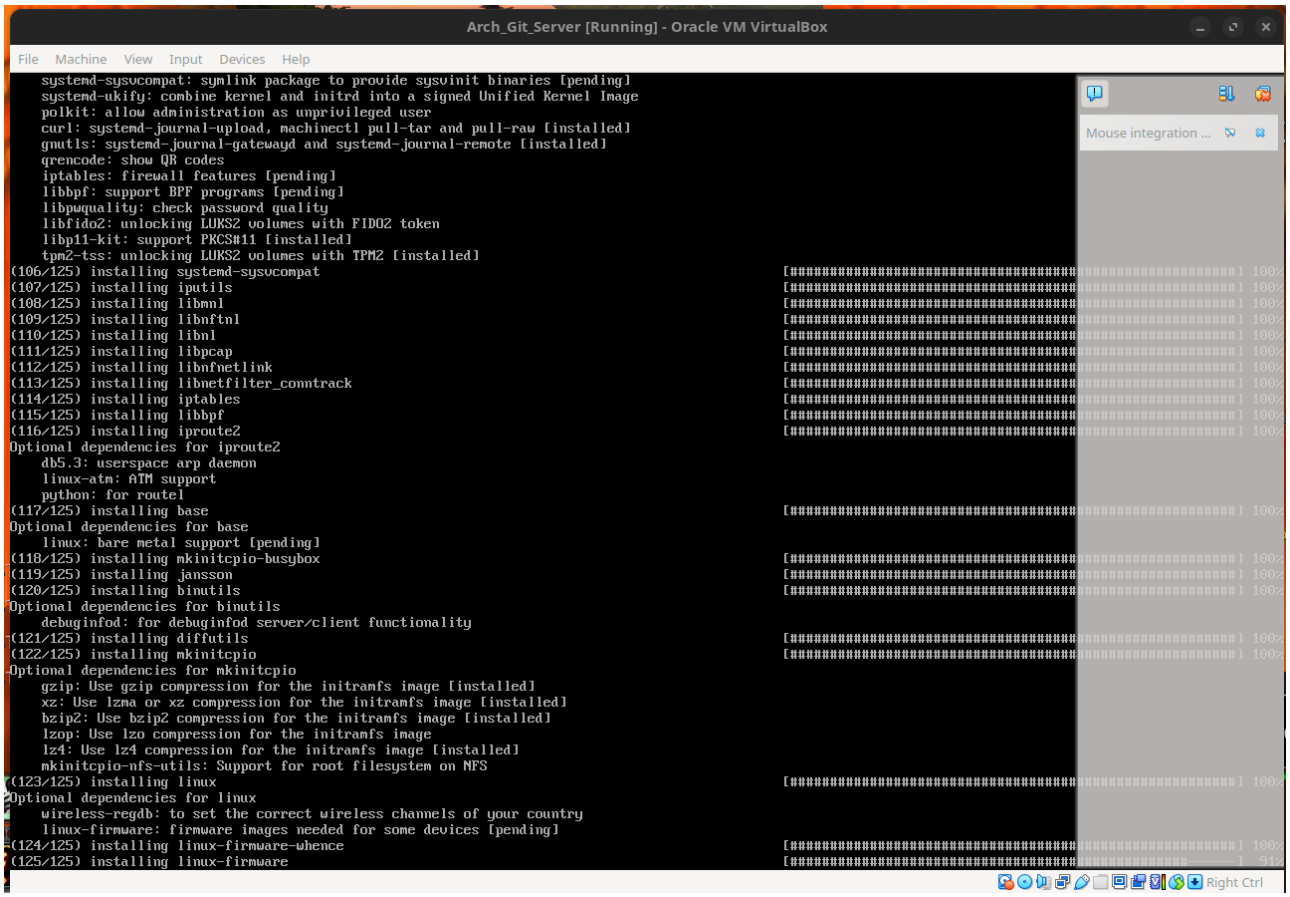
11. Format the partitions. For the main partition run this command “mkfs.ext /dev/sda1” then for the swap run “mkswap /dev/sda2” (note that the dev directory may be different depending on device) (pic 4)

Pic 4



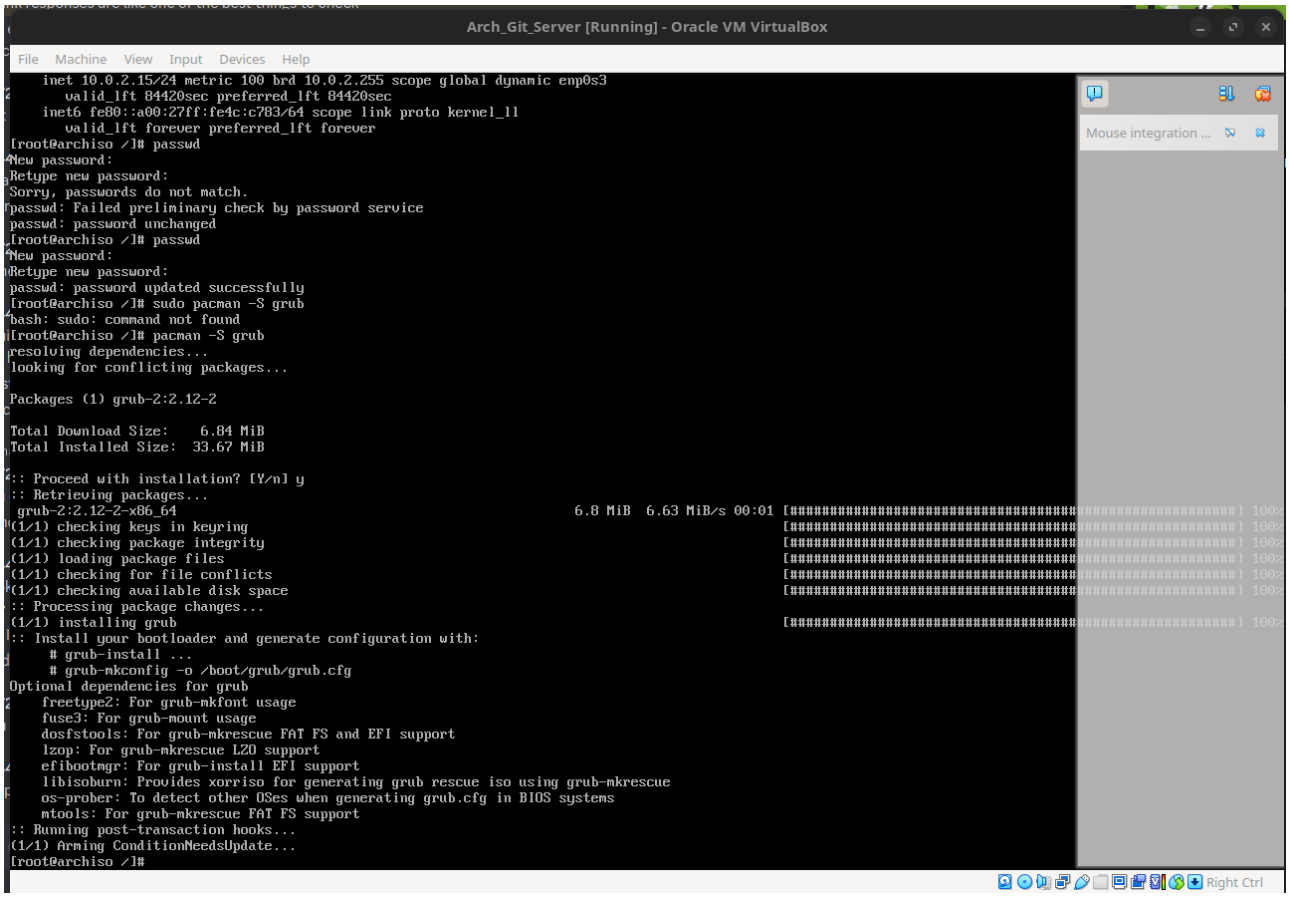
12. Mount the newly created partition on the /mnt directory. This is done by running this command “mount /dev/sda1 /mnt”
13. Enable the swap volume with this command “swapon /dev/sda2”
14. Install the base package, Linux kernel, and firmware to /mnt by running this command “pacstrap -K /mnt base linux linux-firmware”(pic 5)

Pic 5



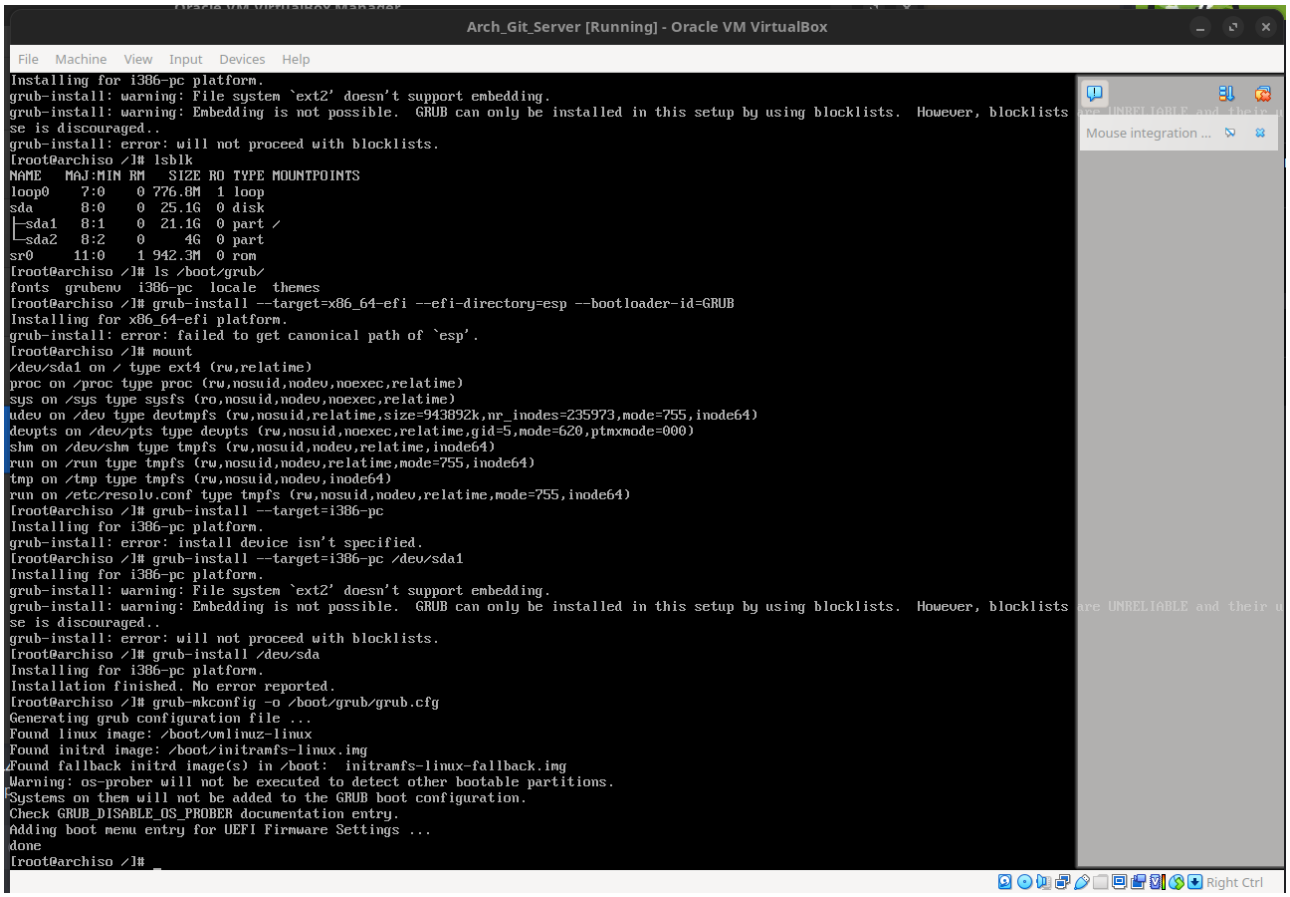
15. Generate a fstab file and place at /mnt/etc/fstab with this command “genfstab -U /mnt >> /mnt/etc/fstab” (The -U option means to create the fstab with uuids).
16. Change root into the new system via this command “arch-chroot /mnt”
17. Install network management software (dhcpcd) via this command “pacman -S dhcpcd”
18. Set the root password by running this command “passwd”
19. Install the Grub 2 package via this command “pacman -S grub”(pic 6)

Pic 6



20. Run this command “grub-install /dev/sda” this installs grub onto the sda device (pic 7)

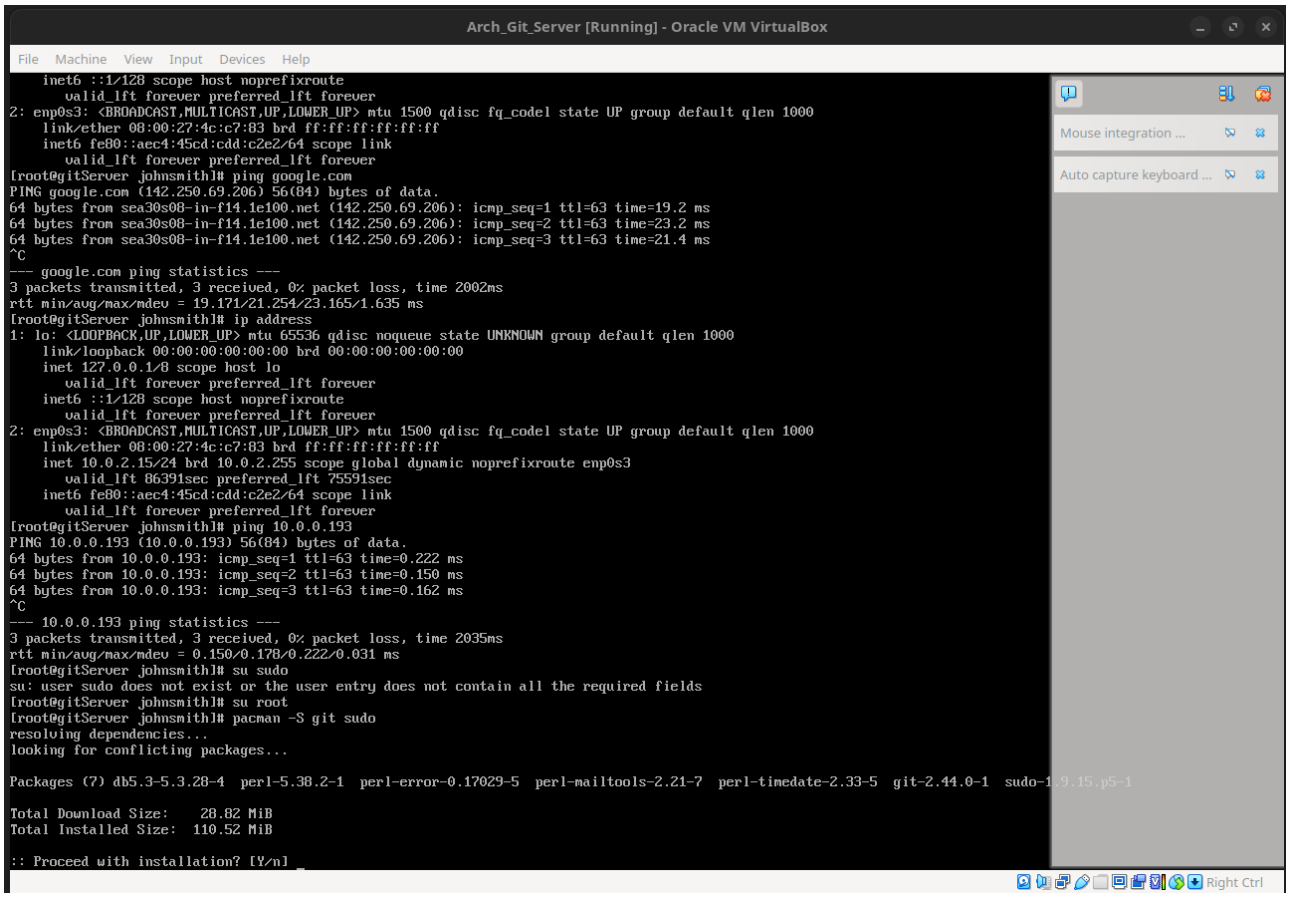
Pic 7




```
Installing for i386-pc platform.
grub-install: warning: File system `ext2' doesn't support embedding.
grub-install: warning: Embedding is not possible.  GRUB can only be installed in this setup by using blocklists.  However, blocklists
se is discouraged..
grub-install: error: will not proceed with blocklists.
[root@archiso /]# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0    7:0    0 776.8M  1 loop
sda      8:0    0  25.1G  0 disk
├─sda1   8:1    0  21.1G  0 part /
└─sda2   8:2    0     4G  0 part
sr0     11:0    1 942.3M  0 rom
[root@archiso /]# ls /boot/grub/
fonts  grubenv  i386-pc  locale  themes
[root@archiso /]# grub-install --target=x86_64-efi --efi-directory=esp --bootloader-id=GRUB
Installing for x86_64-efi platform.
grub-install: error: failed to get canonical path of `esp'.
[root@archiso /]# mount
/dev/sda1 on / type ext4 (rw,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sys on /sys type sysfs (ro,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=943892k,nr_inodes=235973,mode=755,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
shm on /dev/shm type tmpfs (rw,nosuid,nodev,relatime,inode64)
run on /run type tmpfs (rw,nosuid,nodev,relatime,mode=755,inode64)
tmp on /tmp type tmpfs (rw,nosuid,nodev,inode64)
run on /etc/resolv.conf type tmpfs (rw,nosuid,nodev,relatime,mode=755,inode64)
[root@archiso /]# grub-install --target=i386-pc
Installing for i386-pc platform.
grub-install: error: install device isn't specified.
[root@archiso /]# grub-install --target=i386-pc /dev/sda1
Installing for i386-pc platform.
grub-install: warning: File system `ext2' doesn't support embedding.
grub-install: warning: Embedding is not possible.  GRUB can only be installed in this setup by using blocklists.  However, blocklists
se is discouraged..
grub-install: error: will not proceed with blocklists.
[root@archiso /]# grub-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
[root@archiso /]# grub-mkconfig -o /boot/grub/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-linux
Found initrd image: /boot/initramfs-linux.img
Found fallback initrd image(s) in /boot:  initramfs-linux-fallback.img
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done
[root@archiso /]#
```

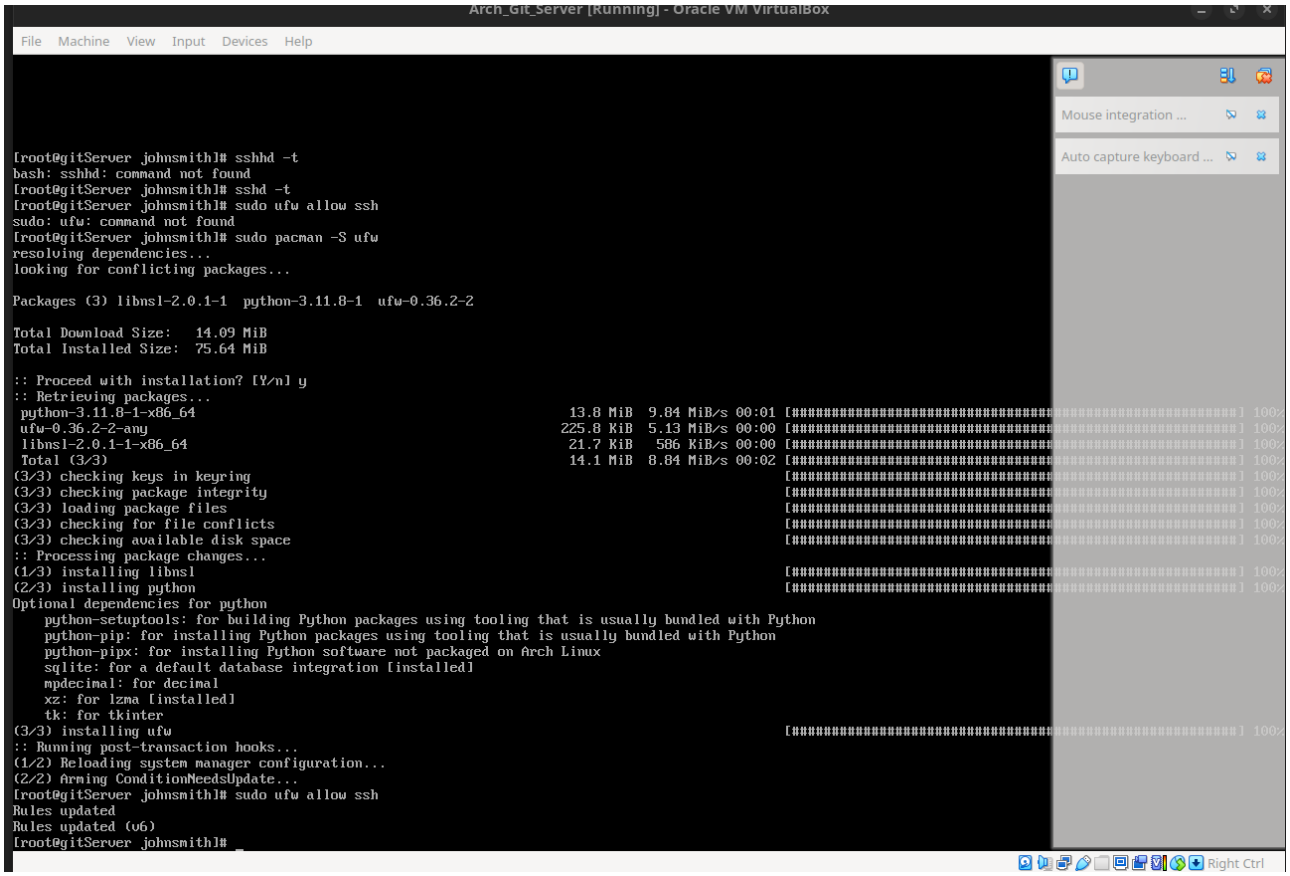
21. The system is now ready to be rebooted first run the command “exit” to exit the chroot environment then run the “reboot” command to reboot the machine.
22. Once the machine has booted into the OS install git via this command “pacman -S git” (pic 8)

Pic 8



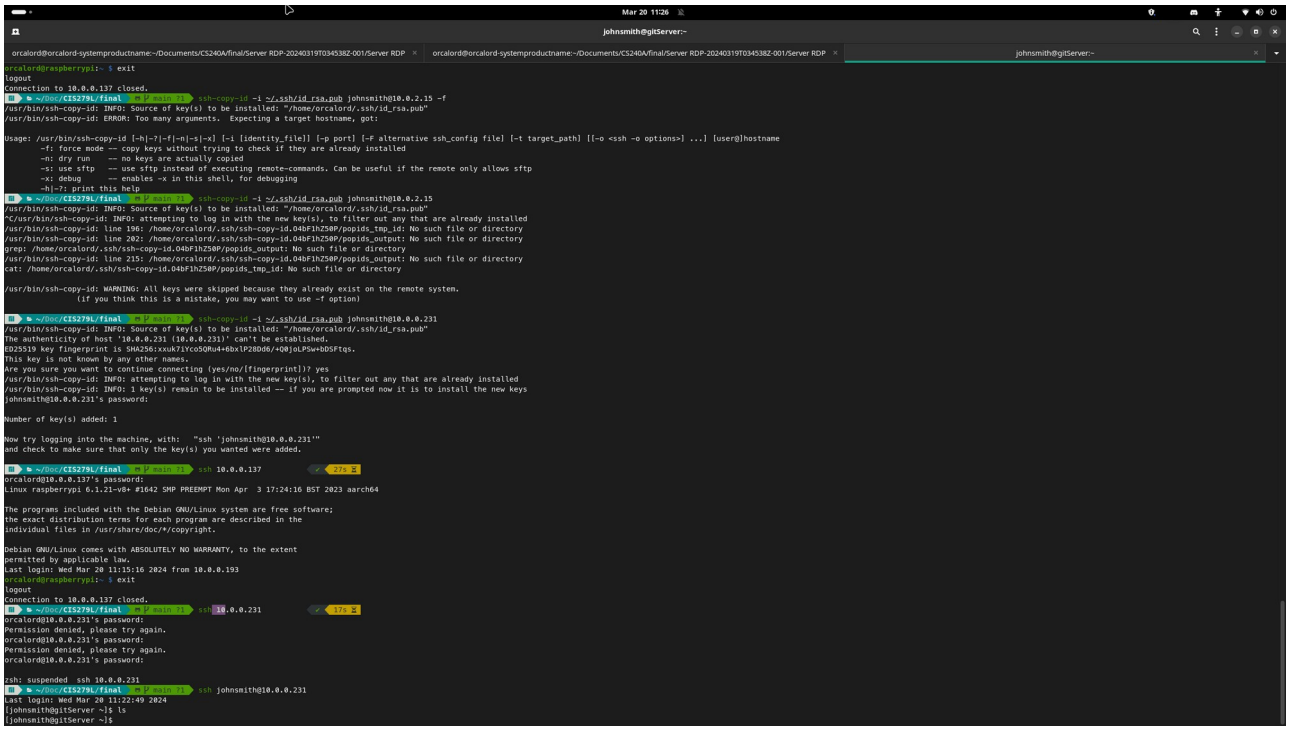
23. Install openssh and ufw via this command “pacman -S openssh ufw” (pic 9)

Pic 9



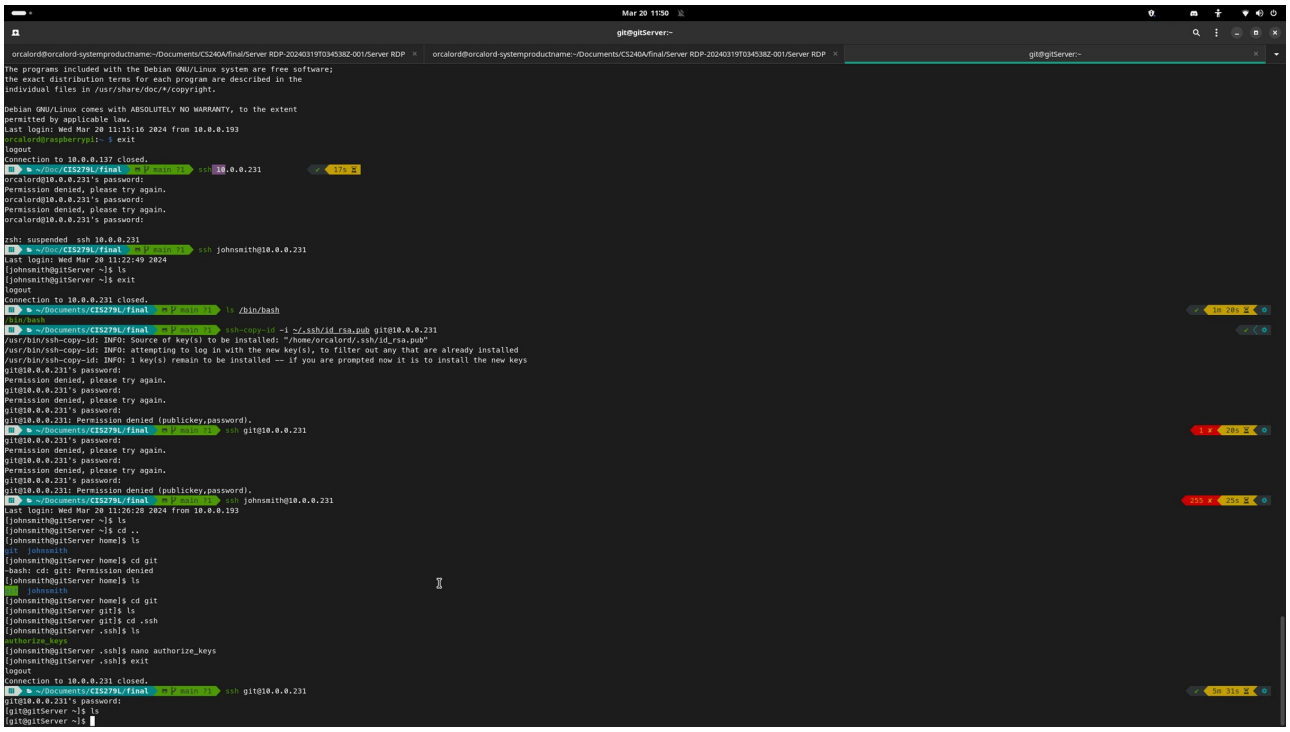
24. Use ufw to open ssh port on the server via this command “sudo ufw allow ssh”
25. enable ssh server with this command “sudo systemctl enable sshd”
26. create user called git via this command “useradd -m -s /bin/bash git”
27. switch to the git user via this command “su git”
28. create a .ssh directory in the home directory via this command “mkdir .ssh” then create a file called authorize_keys with this command “touch .ssh/authorize_keys”
29. On the client generate a rsa pub and private key pair for the ssh via this command “ssh-keygen -t rsa”
30. Copy the public key to the server from the client via this command “ssh-copy-id -i ~/.ssh/id_rsa.pub git@10.0.0.231” (pic 10)(Do note that the user in pic 10 is johnsmith, but the same step applies to the git user)

Pic 10



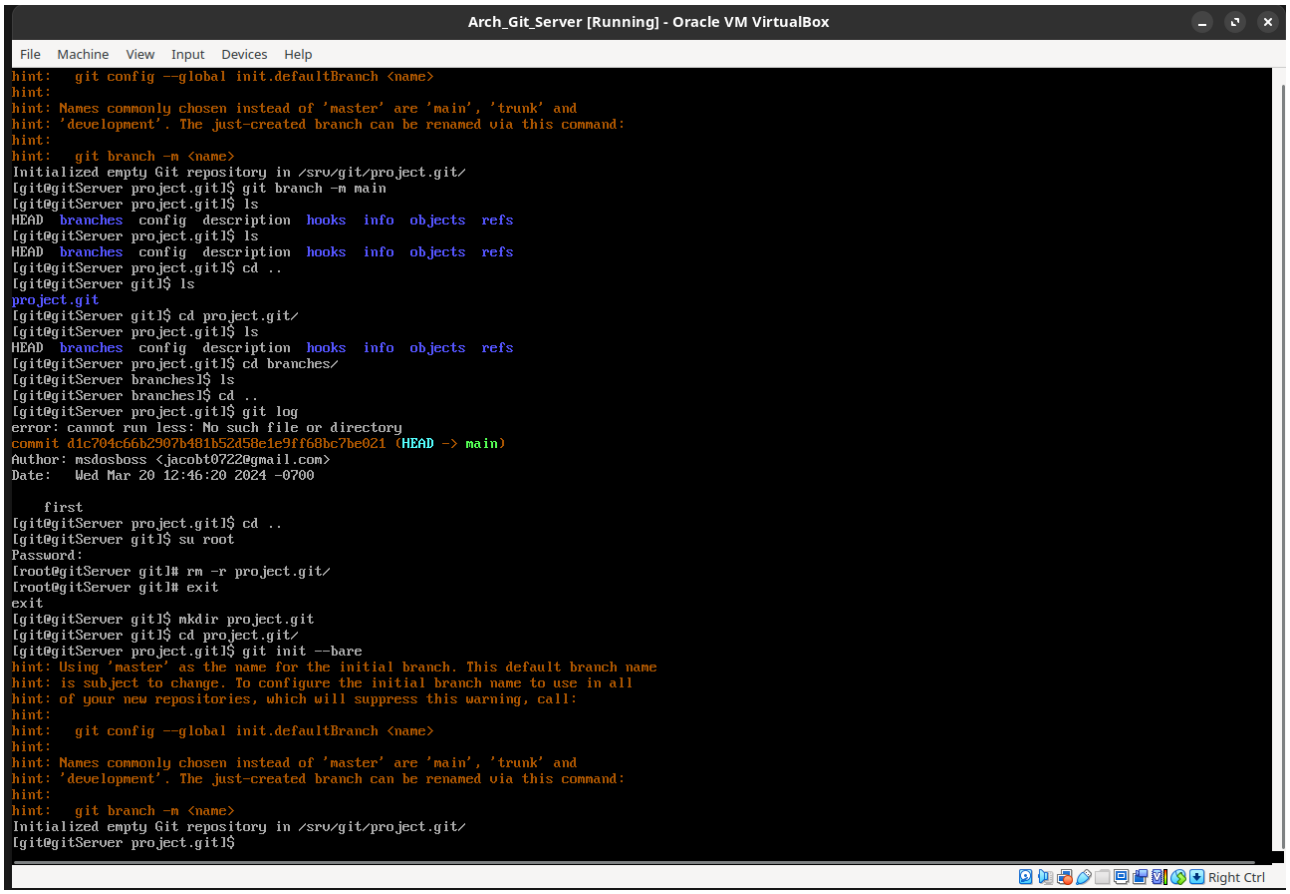
31. Confirm ssh connection to server from client (pic 11)

Pic 11



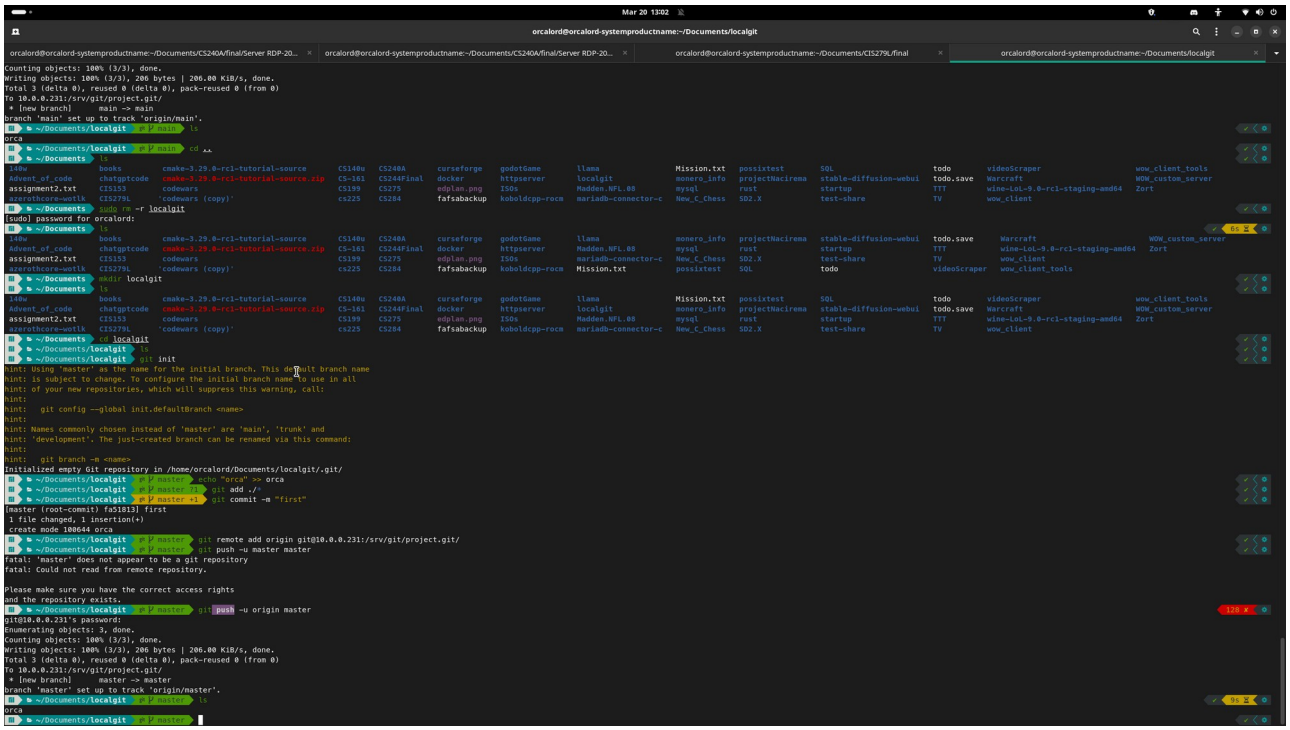
32. On the server create a folder at /srv/git with this command “mkdir /srv/git” then give the git user ownership of the file via this command “chown git /srv/git/” (From steps 32-35 I used this git documentation https://git-scm.com/book/en/v2/Git-on-the-Server-Setting-Up-the-Server)
33. Then run “cd /srv/git” after that create a directory call project.git via this command “mkdir project.git”
34. Run “cd project.git” then create a git repo via this command “git init –bare” (pic 12)

Pic 12



35. Set up a git repo on the client and push it to server via these commands “git init”,”echo “first” >> README.md”,”git add ./*”,”git commit -m ‘first’”,”git remote add origin git@10.0.0.231:/srv/git/project.git”,”git push -u origin master” (Note that github by default calls its branches main, but the git program does it as master) (pic 13)

Pic 13



36. Lastly I will demonstrate that it is working by clone the repo on to a different machine via this command “git clone git@10.0.0.231:/srv/git/project.git” (Note that I did have to set up the public and private keys for the second machine)(pic 14)