arch_documentation

1. Pre-installation

1.1 Acquire an installation image

- 1. Go to the following link: https://archlinux.org/download/
- 2. Scroll down to the United States download links under the HTTP Direct Downloads section
- 3. Choose one of the two mit.edu links

1.2 Verify signature

Once the file from 1.1 has been downloaded, use the following process to make sure the file downloaded correctly (otherwise, redownload):

- 1. Hit Start+R, type in PowerShell, and hit Enter
- Type Get-Filehash followed by a space (Note: instructions found at <a href="https://docs.precisely.com/docs/sftw/spectrum/ProductUpdateSummary/ProductUpdat
- 3. Open the File Explorer, navigate to Downloads, and find the Arch Linux .iso file
- 4. Drag the Arch Linux .iso file to the PowerShell window and hit enter
- 5. Once the hash has finished running, compare the first ten numbers and the last ten numbers of the result to the first ten numbers and the last ten numbers found under Checksums in the HTTP Direct Downloads section of https://archlinux.org/download/

1.3 Prepare an installation medium

This step is not necessary due to the use of a Virtual Machine instead of a direct installation of the iso

1.4 Boot the live environment

- Once the VM has been created, go to the following filepath:
 C:\Users\carls\OneDrive\Documents\Virtual Machines\Arch Linux VM Name
- Right-click on the .vmx file (which will have the icon), mouse-over the 'Open with' option, and select Notepad
- 3. In the Notepad document, under the line that says something to the effect of '.encoding = "windows-1252"', paste the following: firmware="efi". Save and close the notepad

document

4. Turn on your VM. Once your VM is running, select the top option (it should say something like 'UEFI medium install x64'). This will set up your ArchLinux properly

1.5 Set the console keyboard layout

Unnecessary, provided all previous steps have been followed to the letter, the default keyboard (US) should already be set

1.6 Verify the boot mode

Type the following command into the console, then hit enter: 1s /sys/firmware/efi/efivars

If there are no errors, your Arch VM installation has booted to UEFI like it should. Otherwise, the system may have booted in BIOS mode

1.7 Connect to the internet

Given that a VM is being used to host the ArchLinux installation used for this project, it is already connected to the internet. The VM is connected through the computer's wi-fi and ArchLinux thinks that it is connected through a wired connection. Thus, neither wi-fi setup, nor an Ethernet cable are necessary at this time.

1.8 Update the system clock

Run the command timedatect1 status to ensure that date and time are synced. While the time zone will likely be incorrect, all other facets of date and time are now synchronized

1.9 Partition the disks

- 1. Use the command fdisk -1 to list the block devices that the disks available to Arch are assigned to. Pick out the Block Device that you want Arch to be installed on and remember it for the next step.
- 2. Use the command fdisk /dev/the_disk_to_be_partitioned replacing the_disk_to_be_partitioned with the Block Device name picked out earlier. The command line should now say the following: Command (m for help):
- 3. Type n and hit enter to create a new partition. Type p and hit enter to make it a primary partition. Hit enter to select the given default as a partition number. Hit enter again to select the given default as the First Sector for the partition. Type +300M and hit enter to make 300 megabytes the Last Sector for the partition. 300 megabytes is the minimum for an efi system partition, which is what this partition will be used for.

4. You do not need to have a swap partition. Instead, you are going to do the process in step 3 again, but with some changes. Type n and hit enter to create a new partition. Type p and hit enter to make it a primary partition. Hit enter to select the given default as a partition number. Hit enter again to select the given default as the First Sector for the partition. Hit enter a third time to select the given default, which should be the remainder of the drive space allotted for Arch, as the Last Sector for the partition. This partition will be used as the root partition.

1.10 Format the partitions

- 1. For the first partition that was created (the efi partition), type the following command:

 mkfs.fat -F 32 /dev/efi_system_partition
- 2. For the second partition that was created (the root partition), type the following command: mkfs.ext4 /dev/root_partition

1.11 Mount the file systems

Type the following command: mount /dev/root_partition /mnt.

Then, because we are using EFI instead of BIOS, mount the efi partition using the following command: mount --mkdir /dev/efi_system_partition /mnt/boot

2. Installation

2.1 Select the mirrors

Given that I currently don't have enough knowledge about mirrors to confidently make accurate or useful changes to the mirrors file, I will be leaving it as is

2.2 Install essential packages

Run the following command to install the base package: pacstrap -K /mnt base linux linux-firmware

Additionally, for sound functionality (sof-firmware), dhcp (dhcpcd), man, and a text editor (I use nano), run the following command four times, replacing package_name each time with the name of the package: pacman -S package_name

3. Configure the System

3.1 Fstab

Generate an fstab file using the following command: genfstab -L /mnt >> /mnt/etc/fstab

Note: both -U and -L can be used in the command, -U defines by UUID and -L defines by labels

Check the file for errors by using the following command: cat /mnt/etc/fstab

3.2 Chroot

Change root into the newly created system with the following command: arch-chroot /mnt

Stay changed into the root for the remainder of the Arch install setup process

3.3 Time Zone

Set the system time zone with the following command:

```
ln -sf /usr/share/zoneinfo/America/Chicago /etc/localtime
```

America is what is used because you live in the United States, and Chicago is a representation of Central Daylight Time. If you are using Arch in another time zone, you would replace America and Chicago with the relevant Region and City

Run the following command to generate /etc/adjtime: hwclock -systohc

3.4 Localization

- 1. Open the filepath /etc/locale.gen with your text editor. Remove the # before the line that has the following text: en_US.UTF-8 UTF-8. Save the changes to the file and exit the text editor
- 2. Run the following command to generate the locales: locale-gen
- 3. Run the following commands: cd /etc/ then touch locale.conf then nano locale.conf (assuming you are using the nano text editor)
- 4. In the text editor, type LANG=en_US.UTF-8 then save and exit

3.5 Network configuration

- 1. While remaining cd'ed into the /etc/ folder, use the following command: touch hostname
- 2. Use your text editor to start editing the new hostname file, then type in the hostname you want to use as your hostname, then save and exit

3.6 Initramfs

N/A for this use case

3.7 Root password

Use the following command to set a password for root: passwd. You will have to type the password you want twice in a row to make sure you typed it correctly

3.8 Boot loader

Chosen boot loader: GRUB GRUB installation steps:

- 1. Enter the following command to install the grub package: pacman -S grub. Type Y and hit enter when shown the following prompt: Proceed with installation? [Y/n]
- 2. Repeat step 1 with the package called efibootmgr
- 3. Run the following command to properly install grub:

 grub-install --target=x86_64-efi --efi-directory=/boot --bootloader-id=GRUB
- 4. Run the following command to generate /boot/grub/grub.cfg: grub-mkconfig -o /boot/grub/grub.cfg

4. Reboot

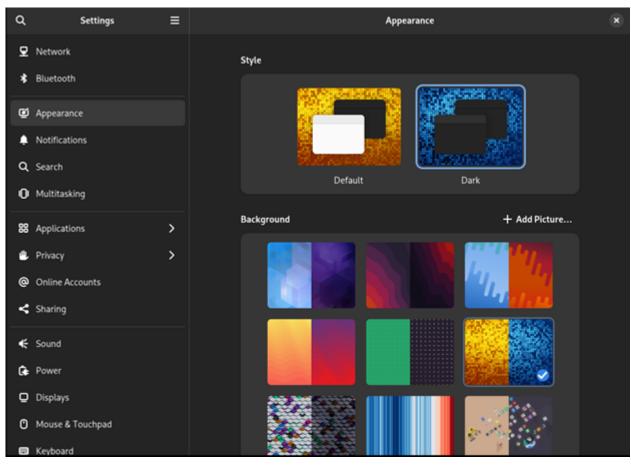
Run the exit command to exit out of the chroot, then run the reboot command

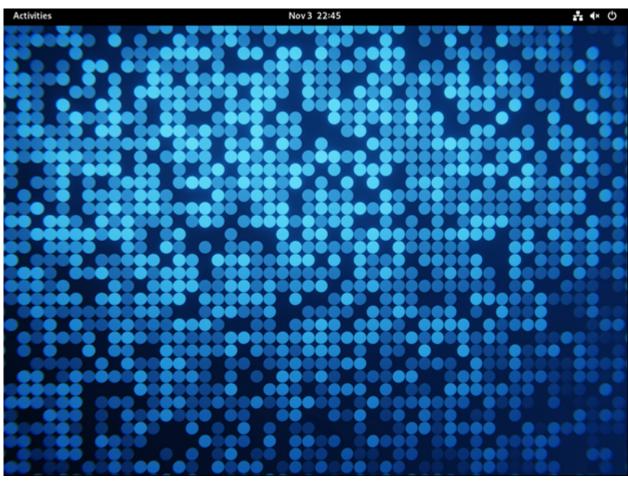
5. Post-installation

Note: before anything can be done with the internet, it must be enabled first. I would recommend running systemctl enable name_of_service on the following services to get the internet working: dhcpcd.service, systemd-networkd.service, and system-resolveed.service. This will make sure that those three programs are active following startup.

From this point on, documentation will occur in the first-person

I chose GNOME as my desktop environment of choice, and installed it with pacman -S gnome, whenever it asked for me to make a choice between certain packages I just hit enter to choose the defaults as I am not overly familiar with GNOME (let alone any other Linux desktop environment). The VM then installed 544 packages. I proceeded to enable gdm.service and reboot the VM. Now, thee GUI/Desktop Environment pops up when the VM boots. I logged into the root account and added accounts for both myself and 'codi' and set the password for 'codi' as instructed. I also changed the Desktop Environment to Dark Mode and chose a background picture.





I installed ssh and remoted into the gateway given to me in the assignment description as shown below

```
[root@archlinux ~]# pacman -S openssh
warning: openssh-9.1p1-1 is up to date -- reinstalling
resolving dependencies...
looking for conflicting packages...
Packages (1) openssh-9.1p1-1
Total Installed Size: 4.22 MiB
Net Upgrade Size: 0.00 MiB
:: Proceed with installation? [Y/n] Y
(1/1) checking keys in keyring
                                                [########## 100%
(1/1) checking package integrity
                                                [################# 100%
(1/1) loading package files
                                                [############### 100%
(1/1) checking for file conflicts
                                                [########## 100%
(1/1) checking available disk space
                                               [################# 100%
:: Processing package changes...
(1/1) reinstalling openssh
                                               [############## 100%
:: Running post-transaction hooks...
(1/3) Reloading system manager configuration...
(2/3) Creating temporary files...
(3/3) Arming ConditionNeedsUpdate...
[root@archlinux ~]#
```

```
[root@archlinux ~]# ssh 129.244.245.111
root@129.244.245.111's password:
```

NOTE: No matter how much googling I did, I could not figure out how to permanently change the color output of ksh. As such, I'm just going to take the lowered grade on this assignment. I can't find any kind of .kshrc or .profile file in my home directory like the answers on Google says I should. As such, I gave up on trying to colorize my shell. The same issue also prevented me from aliasing KornShell.

Below are examples of the aliases I created:

```
[root@archlinux jetman24]# alias lah="ls -lah"
[root@archlinux jetman24]# lah
total 80K
drwx----- 14 jetman24 jetman24 4.0K Nov 4 17:33 .
drwxr-xr-x 4 root
                   root 4.0K Nov 3 21:23 ..
-rw----- 1 jetman24 jetman24 210 Nov 4 17:35 .bash_history
rw-r--r-- 1 jetman24 jetman24 21 Jan 8 2022 .bash_logout-
rw-r--r-- 1 jetman24 jetman24 141 Jan 8 2022 .bashrc
drwx----- 13 jetman24 jetman24 4.0K Nov 3 22:45 .cache
drwx----- 11 jetman24 jetman24 4.0K Nov 3 22:48 .config
-rw----- 1 jetman24 jetman24 20 Nov 4 17:14 .lesshst
drwx----- 3 jetman24 jetman24 4.0K Nov 3 22:38 .local
rw----- 1 jetman24 jetman24 896 Nov 4 17:54 .sh_history
drwxr-xr-x 3 jetman24 jetman24 4.0K Nov 4 17:33 .var
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Desktop
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Documents
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Downloads
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Music
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Pictures
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Public
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Templates
drwxr-xr-x 2 jetman24 jetman24 4.0K Nov 3 22:38 Videos
[root@archlinux jetman24]#
```

```
[root@archlinux ~]# alias cdetc="cd /etc"
[root@archlinux ~]# cdetc
[root@archlinux etc]#
```

```
[root@archlinux jetman24]# alias cdbin="cd /bin"
[root@archlinux jetman24]# cdbin
[root@archlinux bin]# cd
```