

Install Arch Linux on a Macbook Air

A set of tips for installation and post install

Ronald (Ryy) Glenn Thomas

2024-11-09

Table of contents

1	Introduction	2
2	Download	2
3	Install Arch on a Macbook Air	3
4	Install Base Arch Components	3
4.1	Connect to target machine with ssh	3
4.2	Begin install process	4
5	Interface configuration	6
6	Additional Software setup	6
6.1	setup YAY	6
7	Appendix 1. Script to set up links from local Home to Dropbox	7
8	Appendix 2	8



1 Introduction

The specific goal of this “DIY” project is to refurbish a 2016 `macbook air` laptop with a contemporary `linux` operating system.

The OS we’ll focus on for this post is `Arch Linux`, a rolling `Linux` distribution. Why `Arch`? Well, we’re looking for a lightweight, fast installing distro that has access to the `AUR` repository of apps. There are many other considerations that can go into choosing a `linux` distribution, but for our purposes, this is the main one.

2 Download

To get started, acquire a copy of the `Arch Linux` distribution. The simplest way to do this is to download the latest `ISO` image file and “burn” it onto a `USB` drive. (We’ll use an `Apple 2021 Macbook Air` laptop running `macos Sonoma` to facilitate the download and writing onto a 60 GB `USB` drive.) At this writing the latest `iso` file is version 2024.08.01 which is based on `linux` kernel: 6.10.2. The `ISO` Size is 1.1 GB. We’ll use a torrent file (`archlinux-2024.07.08-x86_64.iso.torrent`) to download the `iso` file using the torrent client `Transmission`.

Also download the associated `sha256sum.txt` file.

To start the download install the mac app [Transmission](#) and add the torrent file.

Once download is complete check the integrity of your local `ISO` file, generate its `SHA256` checksum and compare it to the content of the `sha256sum.txt` file:

```
> sha256sum archlinux-2024.07.01-x86_64.iso
```

compare to `SHA256` sums from the download site. In our case:

398dceea2d04767fbb8b61a9e824f2c8f5eacf62b2cb5006fd63321d978d48bc

3 Install Arch on a Macbook Air

The target machine is a 2016 13-inch MacBook Air with one Thunderbolt 3 port.

Next transfer the `iso` file to a USB flash drive using one of several methods. On `macos` we suggest using the app `balanaEtcher`. You can download `balanaEtcher` [here](#).

Insert the USB flash drive into the target `macbook` and reboot. Hold the `ALT` key while the machine reboots and you'll be presented with a screen offering boot drive options. Select the icon for the USB drive. A `grub` menu will appear.¹

From the `Grub` menu select `Arch Linux install medium (x86_64, UEFI)` and the `arch` install program will start.

¹ **GNU GRand Unified Bootloader (GRUB)**. “When your Linux operating system starts up, GRUB is the first program that runs. It loads the kernel of the operating system, and then the kernel loads the rest of the operating system, including the shell, the desktop environment, and other operating system features.” [codecademy.com](https://www.codecademy.com)

4 Install Base Arch Components

To allow cut and paste from `macos` to target laptop connect on your local network via `ssh`.

4.1 Connect to target machine with `ssh`

configure WIFI

- `bash> iwctl`
-
- `idw> device list (optional) (assume device is wlan0)`
- `idw> station wlan0 scan (optional)`
- `idw> station wlan0 connect rgtnet2`
- passphrase for `rgtnet2`
- `idw> exit`

- (assume the local IP address is 10.0.1.176)
- Set a password for root user. You'll need it to log in.
- bash> passwd
- (enter `z` password)
- New password: z
- Retype new password: z

Now switch over to the mac.

```
ssh root@10.0.1.176
```

4.2 Begin install process

First step:

- partition harddisk:

```
bash> cfdisk /dev/nvme0n1
```

Use interface to create two partitions:

1. EFI type of size 1gb
2. root of size entire rest of disk.
3. write partition to disk

check the partition:

- Format the partitions.
 - EFI disk is fat32
 - Root is ext4

```
# run in three parts
# part 1
mkfs.fat -F32 /dev/nvme0n1p1
mkfs.ext4 /dev/nvme0n1p2
loadkeys mac-us
timedatectl set-ntp true
mount /dev/nvme0n1p2 /mnt
reflector -p https --save /etc/pacman.d/mirrorlist --country US --latest 15\
```

```

--sort rate
pacman -Sy
pacstrap /mnt base linux linux-firmware sudo
genfstab -U /mnt >> /mnt/etc/fstab

# part 2
arch-chroot /mnt

# part 3
ln -sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
sed -i.bak 's/#en_US.UTF-8 UTF-8/en_US.UTF-8 UTF-8/' /etc/locale.gen
locale-gen
echo zz >> /etc/hostname
echo LANG=en_US.UTF-8 >> locale.conf
pacman -S --noconfirm networkmanager intel-ucode grub efibootmgr \
    docker xorg-server xf86-video-intel gnome cinnamon vim sudo openssh \
    zsh base-devel pandoc r firefox git fzf ripgrep zathura

systemctl enable NetworkManager gdm sshd docker
mkdir /boot/efi
mount /dev/nvme0n1p1 /boot/efi
grub-install --target=x86_64-efi --bootloader-id=GRUB --efi-directory=/boot/efi
grub-mkconfig -o /boot/grub/grub.cfg

# next items are interactive
#### in sudoers uncomment # %wheel ALL=(ALL) ALL

vim /etc/sudoers
useradd -m -G wheel -s /bin/bash z
passwd z
# password for root
passwd
exit
umount -l /mnt

```

That's it. The base system is ready to go. Reboot and login with the admin username and password you provided earlier. To choose the Cinnamon desktop environment, select **z** as the user and then click on the selection wheel. The default desktop is Gnome, but an option to switch to Cinnamon is offered.

5 Interface configuration

Set keyboard and trackpad preferences:

- 1) Toggle Reverse scrolling direction. Use Super key (Command on mac) to open menu. Type Touchpad in text box.

Select Mouse and Touchpad

Open Touchpad tab

Toggle Reverse scrolling direction

- 2) Select “Swap Esc and Caps-Lock

Type Keyboard in overview text box.

Select in sequence Keyboard > Layouts > Options > Caps Lock behavior

Select “Swap Esc and Caps-Lock (This is an important setting for vim use.)

- 3) Set shortcuts for close window and toggle maximization

Type Keyboard in overview text box.

- Open Shortcuts > Windows.
- Set Toggle maximization state to Super-F
- Set Close window to Super-Q

6 Additional Software setup

6.1 setup YAY

```
sudo git clone https://aur.archlinux.org/yay-git.git
sudo chown -R z:z ./yay-git
cd yay-git
makepkg -si
yay -Syu
yay -S autojump
```

```
yay -S zsh-autosuggestions-git  
yay -S zotero
```

Make `zsh` the default shell.

```
> chsh -s $(which zsh)
```

Start Dropbox to transfer working environment

```
yay -S dropbox  
dropbox  
# builtin autostart from preferences 8/12/24
```

Dropbox startup process will launch a “Sign in” web page. Login with Dropbox credentials through web page.

Next

Run bash shell script `~/Dropbox/dotfiles/set_up_links.sh` to set up symbolic links (e.g. `ln -s ~/Dropbox/prj ~/prj`). See Appendix 1 below for details.

Set up the shell (`zsh`) per the post [link to set up terminal post]

Install `zotero` using software manager and set up syncing (login: `rgthomas`)

add `vimium` extension to `firefox`

7 Appendix 1. Script to set up links from local Home to Dropbox

```
set_up_links.sh
```

```
#!/bin/zsh

# since the install process creates a .config directory move it temporarily
mv ~/.config ~/.config.tmp

# create links to hidden files from ~/Dropbox/dotfiles directories
ff=( ".zshrc" ".viminfo" ".vimrc" ".local" ".vim" \
      ".vimplugins" ".config" ".Rprofile" )
for P in "${ff[@]}"
do
  echo "create a link for Dropbox/dotfiles version of $P in Home"
  ln -v -s "$HOME/Dropbox/dotfiles/$P" "$HOME/$P"
done

# copy the original ".config" files into new linked .config
cp -R ~/.config.tmp/* ~/.config

# create new directories (links) for working files from Dropbox
dd=( "sandbox" "bin" "docs" "prj" "work" "ssh" "shr" )
for P in "${dd[@]}"
do
  echo "create a link for Dropbox/dotfiles version of echo $P in Home"
  ln -v -s "$HOME/Dropbox/$P" "$HOME/$P"
done
```

8 Appendix 2

Connect to new machine via ssh from mac laptop

First on the new machine (zz)

```
scp -r /Users/zenn/Dropbox/work/teaching/fmph243b/project1 z@10.0.1.178:~
scp -r /Users/zenn/Dropbox/dotfiles/kickstart z@10.0.1.178:~
```

- Install R packages
 - Use


```
cd /usr/lib/R/  
sudo chown -R z:z library  
R -e 'install.packages("pacman", repo="cran.rstudio.com")'  
R -e 'install.packages("rmarkdown", repo="cran.rstudio.com")'  
R -e 'tinytex::install_tinytex()'
```

- Install zotero

```
yay -S zotero
```

Possible Shortcut

Install dropbox first. You could wait for Dropbox to finish installing or you could use scp to copy and run the two shell scripts: `install_app.sh` and `set_up_links.sh` from `~/Dropbox/dotfiles`. These two shells can run in parallel with Dropbox installing.