

# Install Linux Mint on macbook Pro

A set of tips for installation and post install

Ronald (Ryy) Glenn Thomas

2023-06-06

## Table of contents

<b>1 Introduction</b>	<b>1</b>
1.1 Install Mint onto a macbook pro . . . . .	1
<b>2 Next steps...</b>	<b>2</b>
<b>3 Install Ubuntu on Parallels</b>	<b>3</b>
3.1 Script to set up links from local Home to Dropbox . . . . .	4

## 1 Introduction

### 1.1 Install Mint onto a macbook pro

The target machine is a 2016 13-inch MacBook with four Thunderbolt 3 Ports.

Get an iso file from the Mint website. The Linux Mint 21.1 Vera download site is [here](#).

The ISO file is 2.5 GB in size. The download torrent file is [here](#)

Use an app on macbook pro running macos, we suggest **Transmission**, to open torrent file and download “iso”. Takes about 6 minutes via ethernet connection.

Transfer (“burn”) iso file to USB drive. On MacOS we suggest using **balanaEtcher**. Download **balanaEtcher** [here](#)

Insert the USB flash drive into the target mac and reboot. Hold the ALT key to allow booting from the external USB drive.

From the **Grub** menu choose “Start Linux Mint 21.1 Cinnamon 64-bit”

A **linux mint** desktop will appear. There is one icon on the screen labeled “install Linux Mint”. Double click it.

A setup dialog will start.

- First Language,
- then Network,
- then codexes,
- then options for a fresh install or “something else”.
- next Location,
- then name a root user and assign a password.

That's it. Reboot and login.

Use Panda Wireless modem for internet access. Mint 21.1 has **Ralink** RT5372 drivers installed.

Add second monitor. Cinema display 27”.

Open Display menu. Select 2560x1440 for resolution for both monitors. Monitor scale 150%.

Turn on Timeshift to provide hourly snapshots

Set keyboard and trackpad preferences:

- Open “Mouse and Touchpad”. Turn on “Tap to click” and “Reverse scroll”.
- Open “Keyboard” > “Layouts” > “Options” > “Caps Lock behavior” and select “Swap Esc and Caps-Lock”

Update base software.

```
sudo apt-get update ; sudo apt-get upgrade
```

## 2 Next steps...

Basics install utilities **git**, **wget**, **curl**, **zsh**

```
sudo apt update
sudo apt install zsh wget curl git -y
```

Install oh-my-zsh

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh)"
```

Install Dropbox and Texlive-full. The easiest way to install Dropbox and Tex is with the Mint software manager.

Install qutebrowser.

```
sudo apt install qutebrowser
```

Install zotero. Easiest method IMO is to use the Debian package

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/retorque/zotero-deb/master/install.sh)"
sudo apt update
sudo apt install zotero
```

### 3 Install Ubuntu on Parallels

- step one set up parallels tools. Use “Actions” menu to select “install parallels tools. This will point CD/DVD device at the iso image. Open the DVD in”Files”. Open install parallel tool-gui.
- macbook hddrive will be available under share
- Firefox is installed. Add vinium-ff extension.

Idea: Could I use Docker?

This seems like genius:

```
alias nvim="docker run -it -v `pwd`: /mnt/volume --workdir=/mnt/volume anatolelucet/neovim:latest"
alias nvim="docker run -it -v `pwd`: /mnt/volume --workdir=/mnt/volume anatolelucet/neovim:latest"
```

### 3.1 Script to set up links from local Home to Dropbox

```
#!/bin/sh
declare -a ff=(
    ".zshrc"
    ".vimrc"
    ".local"
    ".vim"
    ".config"
)

## now loop through the above array
for i in "${ff[@]}"
do
    echo "$i"
    # or do whatever with individual element of the array
done
for P in "${ff[@]}"
do
    ## skip existing links
    if [ -h "$HOME/$P" ]; then
        echo " ";
        echo $P;
        echo "exists in Home";
    fi
    # move existing dir out of the way
    if [ -e "$HOME/$P" ]; then
        if [ -e "$HOME/$P.bak" ]; then
            echo "want to override $HOME/$P but backup exists"
            continue;
        fi
    fi
    #
    echo "Create a Backup .bak file "
    mv -v "$HOME/$P" "$HOME/$P.bak"
    fi
    #
    echo "create a link for Dropbox/dotfiles version of"
    echo $P;
    echo "in Home";
```

```

        ln -v -s "$HOME/Dropbox/dotfiles/$P" "$HOME/$P"
done
if [ ! -d $HOME/sbx ]
then
    echo "sandbox does not exist. Create link"
    ln -v -s $HOME/Dropbox/sandbox/ $HOME/sandbox;
fi
if [ ! -d $HOME/bin ]
then
    echo "bin does not exist. Create link"
    ln -v -s $HOME/Dropbox/bin/ $HOME/bin;
fi
if [ ! -d $HOME/docs ]
then
    echo "docs does not exist. Create link"
    ln -v -s $HOME/Dropbox/docs/ $HOME/docs;
fi
if [ ! -d $HOME/prj ]
then
    echo "prj does not exist. Create link"
    ln -v -s $HOME/Dropbox/prj/ $HOME/prj;
fi
if [ ! -d $HOME/work ]
then
    echo "work does not exist. Create link"
    ln -v -s $HOME/Dropbox/work/ $HOME/work;
fi
if [ ! -d $HOME/shr ]
then
    echo "shr does not exist. Create link"
    ln -v -s $HOME/Dropbox/shr/ $HOME/shr;
fi

declare -a dd=(
    "sbx"
    "bin"
    "docs"
    "prj"
    "work"
    "shr"

```

```

)

## now loop through the above array
for i in "${dd[@]}"
do
    echo "$i"
    # or do whatever with individual element of the array
done
for P in "${dd[@]}"
do
    ## skip existing links
    if [ -h "$HOME/$P" ]; then
        echo " ";
        echo $P;
        echo "exists in Home";
    fi
    # move existing dir out of the way
    if [ -e "$HOME/$P" ]; then
    if [ -e "$HOME/$P.bak" ]; then
        echo "want to override $HOME/$P but backup exists"
        continue;
    fi
    #
    echo "Create a Backup .bak file "
    mv -v "$HOME/$P" "$HOME/$P.bak"
    fi
    #
    echo "create a link for Dropbox/dotfiles version of"
    echo $P;
    echo "in Home";
    ln -v -s "$HOME/Dropbox/$P" "$HOME/$P"
done

```

1+32