Arch Linux

Green = Command Line Code
Blue = Directory
Purple = Modifying Source Code

Installing Arch : This is easier with an ethernet connection if not type in wifi-menu to set up wifi connection. Once you have that and are in the root shell after boot you want to ping a website to ensure connectivity.

1.) You want to sync your clock with timedatectl set-ntp true

2.) Partitioning drives use lsblk to see connected drives lets say you want to configure the sda disk to have it so you will type cfdisk /dev/sda choose either GPT or DOS now. First partition to make is the boot partition you only need 128M if you are using GRUB. With the partition selected hit b to set boot flag you can set the other partition with the rest of the memory if you do not want a swap partition. You then want to WRITE changes and then QUIT.

3.) Formatting comes next with mkfs.ext4 /dev/sda1 then mkfs.ext4 /dev/sda2

4.) Mounting the root partition with mount /dev/sda2 /mnt then create a directory there with mkdir /mnt/boot and mount boot partition with mount /dev/sda1 /mnt/boot use lsblk again to check work.

5.) Strapping your install with pacstrap /mnt base base-devel linux linux-firmware vim

6.) Generating a fstab file making sure to use genfstab -U /mnt this way it is linked to the drives unique identifier hex string instead of the folder path.

7.) Now to chroot into the install itself with arch-chroot /mnt /bin/bash changing from the usb drive boot media into arch install giving you access to pacman.

8.) Time to install a network manager and a boot loader mine will be GRUB to do so you do pacman -S networkmanager grub

9.) Now we want the system to start the network manager whenever we boot with systemctl enable NetworkManager

10.) Configuring GRUB with grub-install /dev/sda then to generate the configuration files with grub-mkconfig -o /boot/grub/grub.cfg if output does not say found linux image and initrd image go back and pacstrap on step 5

11.) Set root password with passwd

12.) Gen locale (selecting language) with vim /etc/locale.gen find the right language and uncomment the UTF and ISO and WRITE QUIT out of vim. Then generate with locale-gen set it in locale.conf with vim /etc/locale.conf and type LANG=en-US.UTF-8 (or whatever language you choose) WRITE QUIT.

13.) Host name setup with vim /etc/hostname and name it what you want and WRITE QUIT.

14.) Set time zone with ln -sf /usr/share/zoneinfo/America/New_York /etc/localtime

15.) Exit chroot with exit

16.) Unmounting root and boot with umount -R /mnt

17.) Reboot with reboot

Installing DWM :

1.) Adding a user to the wheel group with useradd -mg wheel josh and set their password with passwd josh

2.) Give sudo access with vim /etc/sudoers and uncomment code under where it says members of group wheel to execute any command. Then WRITE QUIT.

3.) Getting xorg with pacman -S xorg-server xorg-xinit libxinerama libxft

4.) Cloning DWM with git clone https://github.com/Jd200010/dwm

5.) Cloning ST with git clone https://github.com/MentalOutlaw/st

6.) Cloning Dmenu with git clone https://github.com/MentalOutlaw/dmenu

7.) Change into each of those directories and perform make clean install

8.) Installing file manager with pacman -S pcmanfm

9.) Exit out of root with exit and login as your user

10.) In users home directory touch .xinitrc then vim .xinitrc and add exec dwm WRITE QUIT

11.) To start dwm just do startx

12.) *Optional for Intel systems if getting xorg server error trying to startx do sudo pacman xf86-video-intel*

3/10/22 (Downloading from AUR) : Downloaded Librewolf from Arch User Repository (AUR) with git clone https://aur.archlinux.org/librewolf-bin.git then moved into directory and used makepkg -si but saw I needed to validate the pgp key so I used gpg –recv-key (librewolf pgp key) then used makepkg -si again and finally used sudo pacman -U on the .tar.zst file it created

3/11/22 (Donwloading fonts) : Downloaded the Font Awesome font pack then unzipped it with unzip and put it in /usr/share/fonts

3/11/22 (Patching DWM) : From https://dwm.suckless.org/patches/ I first went to the nextprev/ patch and saw the better implementation for it was shiftview.c so I downloaded the file. I then moved the file into the /root/dwm directory. I then used vim to open config.h where I scrolled just above static Key keys[] = { and inserted #include “shiftview.c”. I then went down into static Key keys[] and inserted

```
{ MODKEY,    XK_x,    shiftview,    {.i = +1} },
{ MODKEY,    XK_z,    shiftview,    {.i = -1} },
```

This patch allows me to change my focused window with MODKEY which is alt in dwm and x or z.

3/27/2022 (slstatus) : cloned slstatus to root directory with git clone https://git.suckless.org/slstatus I then did make clean install after doing so I then vimed into config.h and made configs to let me see network traffic, CPU, RAM and date and time after that I went to my .xinitrc in the home directory and added slstatus & at the top above exec dwm to start slstatus each time I startx.