MY Arch install, MY way!

How to install Arch Linux for dummies (me, I'm dummies)

I've had to do this too often, so I might as well write it down.

- Guide

Guide

TODO:

```
- Expand on setting the time/timezone
- more in-depth description of fdisk commands
- add howto for automounting CIFS network share in fstab
```

Sources:

This info mainly comes from the official Arch Install Guide, with bits from related articles:

Grub Info

Sudo info

fdisk info

BEGIN!

1. Download the official Arch install media,, burn to disk or USB, and boot from it. Now is the time to choose between BIOS or UEFI mode, the mode you boot the disk from determines the mode of the resulting OS!

2. Booting the install media brings you to a basic terminal. First step, update mirors and install the latest GPG keys to be able to install packages:

   ```
   pacman -Syu
   pacman -S archlinux-keyring
   ```

3. Check that you have a network connection and internet:

   ```
   ip addr
   ping archlinux.org
   ```

Partition disks

1. identify the disk(s) you wish to install Linux to:

```
fdisk -l
```

2. format disk(s) and add partitions using `fdisk`:

```
fdisk /dev/[diskname]
```

   - Create new "Primary" partitions for each mountpoint.
   - default type is "Linux Filesystem", this is good for root partition
   - for other partitions (like swap and EFI SYS), Partition Type must be changed.
   - Minimum required partitions:
     - 1x "Linux Filesystem" partition to mount to root ("/")
     - 1x "EFI System Partition" if booting in UEFI mode (minimum size 300MB, recommended 1GB)

3. Format partitions you just made:

```
mkfs.ext4 /dev/root_partition
mkfs.fat -F 32 /dev/efi_system_partition
mkswap /dev/swap_partition [if you made a swap partition]
```

Install the OS

1. mount the new root partition to the live CD file system:

```
mount /dev/root_partition /mnt
mount --mkdir /dev/efi_system_partition /mnt/boot
swapon /dev/swap_partition [if you made a swap partition]
```

2. install base OS and useful packages to the new mount:

```
pacstrap -K /mnt base linux linux-firmware nano networkmanager sudo man-db cifs-utils grub efibootmgr os-prober
```

3. generate a mountpoint file and save to new install (requires mount commands in step 1):

```
genfstab -U /mnt >> /mnt/etc/fstab
```

Enter new OS and configure

1. Enter the new install:

   ```
   arch-chroot /mnt
   ```

2. use `timedatectl` to set the correct time zone, enable NTP, and enable the HW clock (RTC). Then run `hwclock` to finalize:

   ```
   hwclock --systohc
   ```

3. edit the file `/etc/locale.gen` and uncomment the line `en_US.UTF-8 UTF-8`, then generate:

   ```
   locale-gen
   ```

4. create the file `/etc/locale.conf` and enter: `LANG=en_US.UTF-8`

5. edit `/etc/hostname` to set the host name

6. set the root password with `passwd`

7. create your user and add it to group `wheel` (for sudo access)

   ```
   useradd -m -G wheel [username]
   passwd [username]
   ```

8. edit the sudoers file `/etc/sudoers` and uncommend the `wheel` line to enable users in the `wheel` group to sudo

9. Install GRUB bootloader to EFI system partition:

   ```
   grub-install --target=x86_64-efi --efi-directory=boot --bootloader-id=[OSNAME]
   ```

10. edit the grub config `/etc/default/grub` to enable finding other OS's on attached disks, uncomment the line:

    `GRUB_DISABLE_OS_PROBER=false`

11. generate GRUB config:

    ```
    grub-mkconfig -o /boot/grub/grub.cfg
    ```

12. enable networking service:

    ```
    systemctl enable NetworkManager.service
    ```

13. exit the chroot via `exit`, then reboot system, removing install media during POST.

Install KDE:

1. once rebooted, log in as new user [user] and install KDE packages:

```
sudo pacman -S xorg plasma plasma-wayland-session kde-applications
```

2. once completed, enable the display manager and restart:

```
sudo systemctl enable sddm.service
sudo reboot
```