## Install

- Download arch linux iso then flash usb

``` wget http://archlinux.uk.mirror.allworldit.com/archlinux/iso/2024.01.01/archlinux-2024.01.01-x86_64.iso sudo dd if=/path/to/archlinux-2024.01.01-x86_64.isotimen83 of=/dev/sdX status=progress ```

- Boot with archlinux iso
- Set the console keyboard layout - did not set
- Setup network - wifi in my case

``` iwctl [iwd]# station wlan0 connect SSID exit or ctrl+d ``` * verify ``` ping archlinux.org ``` 5. Update system clock ``` timedatectl set-timezone Europe/Dublin ```

6. Partition the disks used fdisk to create 2 partitions ``` sda1 ext4 /boot type 83 (and make it bootable) sda2 ext4 LVM - type 8e ``` * encrypt the disk using LUKS: ``` cryptsetup luksFormat /dev/sda2 ``` * encrypt the disk: ``` cryptsetup luksOpen /dev/sda2 luks ```

7. LVM Configuration

* create the physical volume ``` pvcreate /dev/mapper/luks ```

* create volume group ``` vgcreate vg0 /dev/mapper/luks ``` * create the virtual volumes ``` lvcreate -L 8G vg0 -n swap # make it double size of your ram lvcreate -L 40G vg0 -n root lvcreate -l 100%FREE vg0 -n home ``` * make swap partition and format these partitions ``` mkswap /dev/mapper/vg0-swap mkfs.ext4 /dev/mapper/vg0-home mkfs.ext4 /dev/mapper/vg0-root mkfs.ext4 /dev/sda1 ``` 8. Mount partition, create boot and home mount points, turn on swap part ``` mount /dev/mapper/vg0-root /mnt mkdir -p /mnt/{boot,home} mount /dev/mapper/vg0-home /mnt/home mount /dev/sda1 /mnt/boot swapon /dev/mapper/vg0-swap ``` 9. Install base system ``` pacstrap /mnt base vim lvm2 linux linux-firmware ```

* if installation brings some issues with old keys run following and rerun `pacstrap` command ``` pacman -Sy archlinux-keyring ```

10. Generate fstab ``` genfstab -U /mnt » /mnt/etc/fstab ```

11. Change the root path to the new system ``` arch-chroot /mnt ```

12. Timezone ``` timedatectl set-timezone Europe/Dublin hwclock –systohc –utc ``` 13. Set host name ``` echo arch > /etc/hostname ``` 14. Set locale ``` vim /etc/locale.gen # uncomment: # en_US.UTF-8 UTF-8 ``` then run ``` locale-gen ``` then ``` echo LANG=en_US.UTF-8 > /etc/locale.conf echo LANGUAGE=en_US » /etc/locale.conf echo LC_ALL=C » /etc/locale.conf ``` 15. Set root password ``` passwd root ```

16. configure the initram file system to load LVM and LUKS modules before loading the kernel ``` vim /etc/mkinitcpio.conf # and add 'encrypt lvm2' before `filesystem` HOOKS=“base udev autodetect modconf kms keyboard keymap consolefont block encrypt lvm2 filesystems fsck” ``` then run ``` mkinitcpio -P ```

17. Install grub ``` pacman -S grub networkmanager sudo grub-install –target=i386-pc /dev/sda ``` *

edit file `vim /etc/default/grub`, add `lvm` and `cryptdevice=/dev/sda2:luks` to following: ``` GRUB_PRELOAD_MODULES='lvm' GRUB_CMDLINE_LINUX="cryptdevice=/dev/sda2:luks" ``` and run following to regenerate grub config file ``` grub-mkconfig -o /boot/grub/grub.cfg ``` 18. Exit from chroot mode, unmount part and reboot machine ``` exit umount -R /mnt reboot ```

## After install steps

1. Connect wire Eth0 and start Network Manager - then setup wifi ``` systemctl enable –now NetworkManager nmtui ``` 2. Install xorg,lightdm and i3 ``` pacman -S bash-completion xorg-server pacman -S xorg-xinit xorg-xkill pacman -S xterm firefox pacman -S i3-wm i3lock i3status dmenu network-manager-applet pacman -S lightdm lightdm-gtk-greeter systemctl enable lightdm reboot ``` 3. Create user ``` useradd -m karcio passwd karcio ### add user to sudo su - karcio echo "exec i3" » ~/.xinitrc ``` 3. Install sound ``` sudo pacman -S pulseaudio pulseaudio-alsa alsa-utils ``` 4. Install notification ``` sudo pacman -S notification-daemon libnotify ``` then edit `/usr/share/dbus-1/services` ``` sudo vim /usr/share/dbus-1/services/org.freedesktop.Notifications.service ``` and add following lines: ``` [D-BUS Service] Name=org.freedesktop.Notifications Exec=/usr/lib/notification-daemon-1.0/notification-daemon ``` 5. Install additional fonts ``` sudo pacman -S ttf-font-awesome terminus-font ttf-dejavu ``` 6. Install cron ``` sudo pacman -S cronie sudo systemctl enable –now cronie.service ``` 7. Install graph stuff ``` sudo pacman -S feh ``` 8. Install some tools ``` sudo pacman -S git alacritty mc zip unzip ```

https://blog.bespinian.io/posts/installing-arch-linux-on-uefi-with-full-disk-encryption/