# Arch Linux Encrypted LUKS RAID 1 Installation Guide

## Partition RAID 1 Block Devices

Load the correct kernel modules.

1. **modprobe raid1**
2. **modprobe dm-mod**

Partition the drives so that you have a 200MB /boot partition, a 4GB swap partition, and a /root partition that will take up the rest of the disk space.

1. **cfdisk /dev/sda**
2. **Partition Type: Linux RAID (fd00)**
3. **Boot Flag on /boot partition (eg. /dev/sda1)**

## RAID Installation

To install RAID, you need to create a /boot /swap and /root arrays using mdadm. Create the /root array at /dev/md0

1. **mdadm –create /dev/md0 –level=1 –raid-devices=2 /dev/sd[ab]3**
2. **mdadm –create /dev/md1 –level=1 –raid-devices=2 /dev/sd[ab]2**
3. **mdadm –create /dev/md2 –level=1 –raid-devices=2 –metadata=1.0 /dev/sd[ab]1**

After you create a RAID volume with the raid arrays, it will synchronize the contents. You can monitor this progress by freshening this command.

1. **watch -n .1 cat /proc/mdstat**

and information about the RAID array's can be found with this command.

1. **mdadm –misc –detail /dev/md[012]**

Once synchronization is complete the State line should read clean. Each device in the table at the bottom of the output should read spare or active sync in the State column. active sync means each device is actively in the array.

## Encrypted LVM Installation

1. **cryptsetup luksFormat /dev/sda3**

then enter a suitable passphrase for the encrytped device.

1. **cryptsetup open –type luks /dev/sda3 archlvm**

Make the RAID volume accessible to LVM by converting it to a physical volume (PV).

1. **pvcreate /dev/mapper/archlvm**
2. **pvdisplay**

Next create the volume group on the PV.

1. **vgcreate archvg /dev/mapper/archlvm**
2. **vgdisplay**

Now you can create logical volumes which will be accessible through the arch group (archvg)

1. **lvcreate -L 4G archvg -n swap**
2. **lvcreate -L 30G archvg -n root**
3. **lvcreate -l 100%FREE archvg -n home**

Confirm that the lv has properly been created.

1. **lvdisplay**

## Update the RAID Configuration

Since the installer builds the initrd using /etc/mdadm.conf in the target system, you should update that file with your RAID configuration. The original file can simply be deleted because it contains comments on how to fill it correctly, and that is something mdadm can do automatically for you. So let us delete the original and have mdadm create you a new one with the current setup:

1. **mdadm –examine –scan >> /etc/mdadm.conf**

## Finish Installation

Wipe & mount the new created logical volumes.

1. **mkfs.ext4 /dev/mapper/archvg-root**
2. **mkfs.ext4 /dev/mapper/archvg-home**
3. **mkswap /dev/mapper/archvg-swap**
1. **mount /dev/mapper/archvg-root /mnt**
2. **mkdir /mnt/home**
3. **mount /dev/mapper/archvg-home /mnt/home**
4. **mount /dev/sda1 /mnt/boot**
5. **swapon /dev/mapper/archvg-swap**

Download australian mirrorlist for packages.

1. **wget "https://archlinux.org/mirrorlist/?country=AU"**
2. **pacstrap /mnt base base-devel sudo vim**
3. **genfstab -p /mnt >>/mnt/etc/fstab**

   **IMPORTANT!!! Comment out the /boot entry from /etc/fstab to avoid running into a slow boot**
4. **arch-chroot /mnt**
5. **useradd -m archie**
6. **passwd archie**
7. **ln -sf /usr/share/zoneinfo/Australia/Brisbane /etc/localtime**
8. **hwclock –systohc –utc**
9. **passwd**
10. **vim /etc/locale.gen**
11. **locale-gen**
12. **locale >/etc/locale.conf**
13. **vim /etc/hostname**
14. **sudo pacman -S linux linux-firmware mkinitcpio mdadm lvm2 dialog wpa_supplicant netctl dhcpcd**

Then load the amd driver modules and add mkinitcpio hooks for booting.

1. **vim /etc/mkinitcpio.conf**
   MODULES="amdgpu radeon"
   HOOKS="... mdadm_udev keyboard encrypt lvm2 filesystems"
2. **vim /etc/sudoers**
   Edit & Add: archie ALL=(ALL) ALL
3. **mkinitcpio -p linux**

## Configure the BootLoader

Now it's time to add the bootloader!

1. **bootctl –path=/boot/ install**
2. **vim /boot/loader/loader.conf**

clear
default arch
timeout 5
editor 0

1. **vim /boot/loader/entries/arch.conf**

title Arch Linux ENCRYPTED
linux /vmlinuz-linux
initrd /initramfs-linux.img
options cryptdevice=UUID=XXXXXXXXXXX:archlvm
root=/dev/mapper/archvg-root quiet rw
**vim magic: read ! blkid /dev/sda2**

## Update the RAID Configuration

Reupdate the sync of the RAID 1 array.

1. **mdadm –examine –scan >> /etc/mdadm.conf**

## Finish Installation  Reboot

1. **exit**
2. **umount -R /mnt**
3. **reboot**

# i3-gaps Custom Installation Guide

## Connect to internet

After entering the encryption password as well as your user password, now you will have to first connect to the internet.

1. **sudo cp /etc/netctl/example/ethernet-dhcp /etc/netctl/**

Check your ethernet adapter name and enter the interface name in the ethernet-dhcp config.

1. **ip a**
2. **sudo vim /etc/netctl/ethernet-dhcp**
3. **sudo netctl start ethernet-dhcp**
4. **sudo netctl enable ethernet-dhcp**

## Install Ly bootloader

Install the ly bootloader from github

1. **git clone https://github.com/cylgom/ly.git**

Fetch the submodules then compile

1. **make github**
2. **make**

Install ly to system to provide system service file, then enable the service file.

1. **sudo make install**
2. **sudo systemctl enable ly.service**

To login to ly as root, you must comment the following line "auth required pam_securetty.so" in the /etc/pam.d/login file.

## i3-gaps Ricing and Package Installation

To start ricing, install the following packages. Once installed, reboot the system and go through the instructions in creating a default i3 config.

1. **yay -S git i3-gaps xorg termite chromium rofi lxappearance compton-tryone-git ranger python-pywal htop neofetch lightdm-settings tty-clock code arc-gtk-theme spotify zenity ffmpeg-compat-57 pulseaudio pavucontrol node npm ttf-font-awesome networkmanager nm-connection-editor network-manager-applet polybar private-internet-access-vpn python-dbus**
2. **cp -R arch-linux/config/.config ~/** After you have moved all the config files to the current config file in your home directory, click the combination super + shift + r to reload the i3 config. View the changes.

Create your home directories and put the arch-linux git into Downloads. Set the background and setup comtpon kawase blur as well as pywal wallpaper color scheme.

1. **cp ./Downloads/arch-linux/config/wallpaper.jpg ./Pictures/**
2. **pywal -i Pictures/wallpaper.png**

Then custommize vscode themes with the material palenlight theme.Then install private internet access through wget.

1. **wget "https://installers.privateinternetaccess.com/download/pia-linux-1.7-03949.run"**
2. **chmod +x pia-linux-1.7-03949.run**
3. **./pia-linux-1.7-03949.run**

Change system-wide theme with lxappearance to arc-dark.