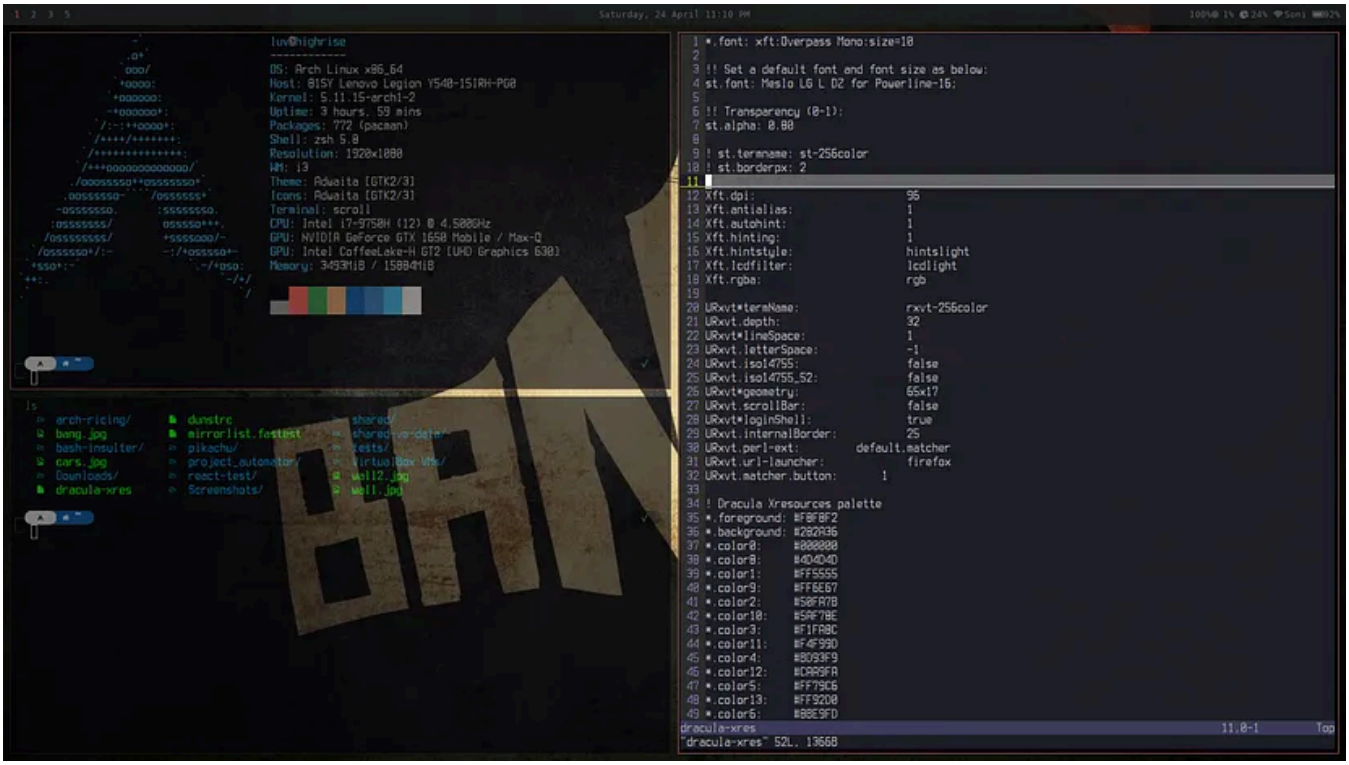
Install Arch Linux & GUI

Basant Soni · Follow
5 min read · Apr 25, 2021

Listen Share ••• More



My Arch Ricing

Arch-Linux is a very popular Linux distribution among Linux power users and developers. It can be a bit daunting if you're installing it for the first time, this is where this article can help you with.

Pre-Installation

Backup Data

Installing Linux may format your disk if done wrong, I'd recommend you to backup important files to a separate disk before installation.

UEFI vs Legacy

Installation steps depend on whether you install arch in UEFI (Unified Extensible Firmware Interface) or Legacy mode. UEFI mode is relatively new and has the following benefits over Legacy mode

- It is compatible with GPT tables
- It supports up to128 primary partitions and disk size up to 2 TB
- It boots faster

To check if your system supports UEFI, boot up into firmware and check for the ability to configure boot mode. If it provides you the option to boot into UEFI mode, it supports UEFI.

Analyze your Disk

> *If your system supports UEFI and you want to switch from Legacy to UEFI with GPT disk, you'll have to format your hard disk to convert it from MBR to GPT.*

- You must have at least 10 GB of free space for Arch.
- Don't worry if you're dual-booting, it won't affect your system in any way. I have 5 OS installed on my laptop for years and it runs perfectly fine.

If you want to install arch in Legacy mode with MBR disk table

1. Make sure you have a maximum of 3 primary partitions since Legacy mode can support up to only 4 primary partitions and 1 primary partition would be created for Arch at the time of installation.
2. I would recommend 3 primary partitions so you can have 3 OS installed (including arch) and 1 extended partition under which you can have logical partitions for data storage.

Make Bootable USB

- Download the Arch Linux ISO image here
- Create a bootable USB using Etcher or Rufus
- Boot your system from the USB

Installation

Internet Connection

Your wireless adapter might be blocked by rfkill, execute the below command to unblock it

```
rfkill unblock wifi
```

Connect to the internet with iwctl

```
iwctl
device list
station wlan0 scan
station wlan0 get-networks
station wlan0 connect <wifi-network>
```

Execute the below command to check if UEFI mode is enabled, if it outputs anything, that means it's enabled and you can install Arch in UEFI mode

```
ls /sys/firmware/efi/efivars
```

Partition the Disk

- To list available partitions and disks, execute the command lsblk, you'll probably notice two drives, USB drive, and HDD/SSD.
- If you use hibernation, then you must add swap because the content of the RAM will be written to the swap partition. This also means that the swap size should be at least the size of the RAM
- If you're installing Arch alongside any other OS in UEFI mode, you must already have an ESP partition, don't create it in that case.
- Disk label type should be gpt if you're in UEFI mode, else dos.
- Partition the disk using any tool like cfdisk, fdisk or parted. I'd suggest cfdisk if you're a beginner.

Assuming disk is sda, create the following partitions

1. ESP FAT32 partition — 512 MB (/dev/sda1)
2. Root Ext4 partition (recommended for beginners) — 10 GB (/dev/sda2)

3. Swap partition — the size of RAM (/dev/sda3)

Format partitions

```
mkfs.fat -F32 /dev/sda1    # EFI
mkfs.ext4 /dev/sda2         # Root
mkswap /dev/sda3           # Swap
```

Mount partitions

```
mount /dev/sda2 /mnt
swapon /dev/sda3
```

Enable Multilib Support

You can also enable Arch Multilib support for the live system by uncommenting the following lines from /etc/pacman.conf file

```
[multilib]
Include = /etc/pacman.d/mirrorlist
```

Install base packages

```
pacstrap /mnt base base-devel linux linux-firmware vim sudo
```

Generate fstab file

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

Chroot to the installed system

It’d change your ISO environment to the root environment of the installed Arch system.

```
arch-chroot /mnt
```

Set Locale

- This is what sets the system language, numbering, date and currency formats for your system.
- Choose and uncomment your preferred encoding languages from /etc/locale.gen file

```
en_US.UTF-8 UTF-8
en_US ISO-8859-1
```

- Generate your system language layout

```
locale-gen
echo LANG=en_US.UTF-8 > /etc/locale.conf
export LANG=en_US.UTF-8
```

Set Timezone

Select your region and time zone.

```
timedatectl list-timezones
timedatectl set-timezone Aisa/Kolkata
# OR
ls /usr/share/zoneinfo/
ln -s /usr/share/zoneinfo/Aisa/Kolkata /etc/localtime
```

Set Hardware Clock

Set time for the system

```
hwclock --systohc --utc
```

Network Configuration

```
echo your-hostname > /etc/hostname
```

```
vim /etc/hosts
# paste below
127.0.0.1       localhost
```

```
::1                     localhost
127.0.1.1       your-hostname
```

- Setup Network-manager

```
pacman -S networkmanager
systemctl enable NetworkManager
```

Enable Multilib Repo

- edit /etc/pacman.conf and uncomment following

```
[multilib]
Include = /etc/pacman.d/mirrorlist
```

- Synchronize and Update

```
pacman -Syu
```

Set up a New User

- Set up a password for the root account and create a new user with Sudo privileges

```
passwd
useradd -mg users -G wheel,storage,power -s /bin/bash your_new_user
passwd your_new_user
chage -d 0 your_new_user
```

- Update the wheel group line from /etc/sudoers file in order to grant root privileges to the newly added user (you can use command visudo to edit the file as well)

```
%wheel ALL=(ALL) ALL
```

Install Bootloader

- For UEFI Systems

```
pacman -S grub efibootmgr dosfstools os-prober mtools
mkdir /boot/efi
mount /dev/sda1 /boot/efi
grub-install --target=x86_64-efi --bootloader-id=GRUB --efi-
directory=/boot/efi
grub-mkconfig -o /boot/grub/grub.cfg
```

- For Legacy Systems

```
pacman -S grub
grub-install /dev/sda
grub-mkconfig -o /boot/grub/grub.cfg
```

Exit

```
exit
umount -a
reboot now
```

This is a minimal installation, you'll be greeted with a console on reboot.

Post-Installation

Install YAY AUR Helper

```
sudo pacman -S git
git clone https://aur.archlinux.org/yay.git
cd yay
makepkg -si --noconfirm
cd .. && rm -rf yay
```

Install some basic tools

```
sudo pacman -S wget vim wpa_supplicant pcmanfm ntfs-3g
yay -S st-luke-git
```

- pcmanfm is a file manager
- ntfs-3g is used to mount NTFS file systems
- st-luke-git is Luke's suckless terminal, I prefer this but you can install any other you like

Install X Window System

```
pacman -S xorg xorg-xinit xorg-server
```

Now you can install a Window Manager like I3 if you want to fully customize your system or a Desktop Environment like Gnome, KDE, etc. I'd recommend Gnome if you're a beginner. We'll cover both.

Install lightdm and I3 Window Manager

```
sudo pacman -S i3-gaps accountsservice openssl lightdm lightdm-gtk-
greeter lightdm-gtk-greeter-settings i3status
```

Edit /etc/lightdm/lightdm.conf as follows

```
greeter-session=lightdm-gtk-greeter
```

Enable and start lightdm

```
sudo systemctl disable gdm -f
sudo systemctl enable lightdm -f
sudo systemctl set-default graphical.target
```

Install GNOME

You don't need to install this if you've already installed a window manager.

```
echo "exec gnome-session" > ~/.xinitrc
sudo pacman -S gnome
```