How to Install Arch Linux [Step by Step Guide]

Ready to get your hands dirty with the for-expert Linux distro? Here's a step by step guide to show you how to install Arch Linux.

Abhishek Prakash
20 Nov 2023 13 min read 337 Comments

ON THIS PAGE

Arch Linux is a general-purpose rolling release Linux distribution that is very popular among DIY enthusiasts and hardcore Linux users.

The default installation covers only a minimal base system and expects the end-user to configure the system by himself/herself.

This is why installing Arch Linux is a challenge in itself but at the same time, it is a learning opportunity for intermediate Linux users.

Fret not, you can easily get Arch Linux up and running if you follow the st .

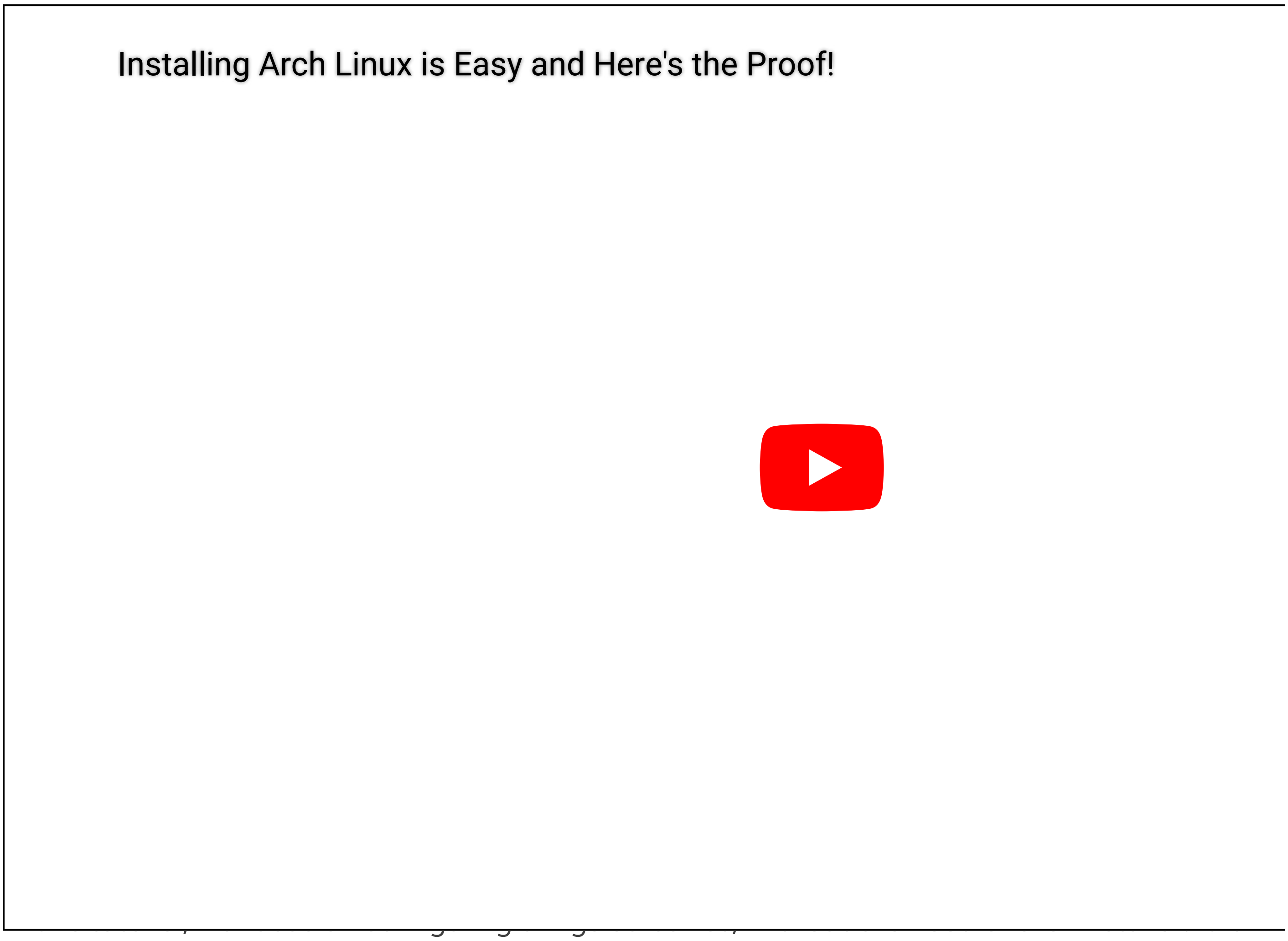
How to install Arch Linux

As of now, there are two ways using which you can install Arch Linux:

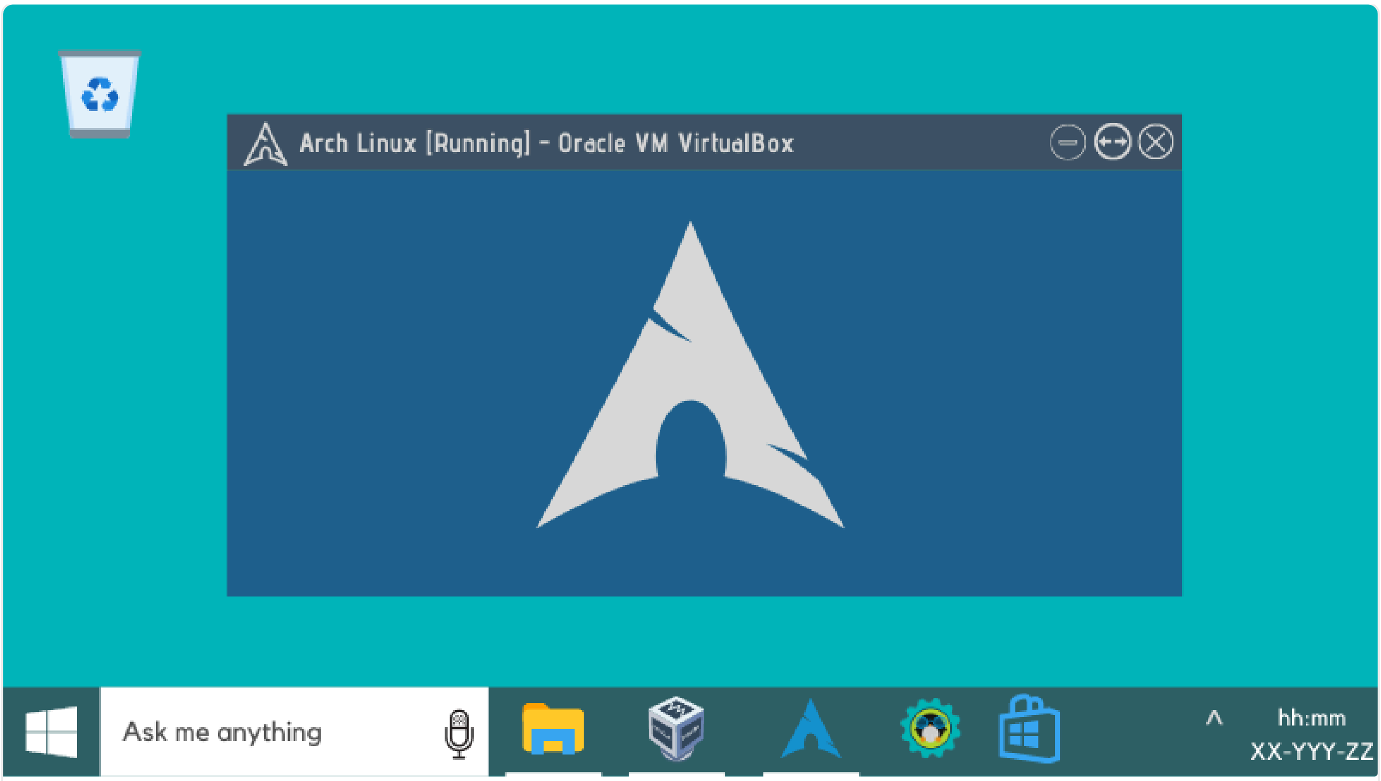
- Without the guided installer
- With the help of Archinstaller (guided installer)

In case you didn’t know, Arch Linux introduced a guided installer in 2021 (a menu based system to help you easily configure things for installation).

The video below demonstrates the use of archinstall guided installation:



potentially more time-consuming, and if you are new to Arch Linux, you might want to refer to our separate tutorial, where we use the guided installer to install Arch Linux using VirtualBox.



Beginner’s Guide to Installing Arch Linux on VirtualBox

Arch Linux is hugely popular in the desktop Linux world. One of the reasons for its popularity is that installing Arch Linux itself is a complicated task. I am not exaggerating. Installing Ubuntu or…

It's FOSS • Ankush Das

The installation steps can differ at some points depending on whether you have a UEFI or legacy BIOS system. Most new systems come with UEFI these days.

I have written it here with a focus on the UEFI system but I’ll also mention the steps that are different for the legacy BIOS systems.

🚧 The method discussed here **wipes out existing operating system**(s) from your computer and installs Arch Linux on it. So if you are going to follow this tutorial, make sure that you have backed up your files, or else you’ll lose all of them. You have been warned.

But before you see how to install Arch Linux from a USB, please make sure that you have the following requirements:

Requirements for installing Arch Linux:

- An x86_64 (i.e. 64 bit) compatible machine
- Minimum 512 MB of RAM (recommended 2 GB)
- At least 2 GB of free disk space (recommended 20 GB for basic usage with a desktop environment)
- An active internet connection
- A USB drive with a minimum 2 GB of storage capacity
- Familiarity with the Linux command line

Once you have made sure you have all the requirements, let’s install Arch Linux.

Step 1: Download the Arch Linux ISO

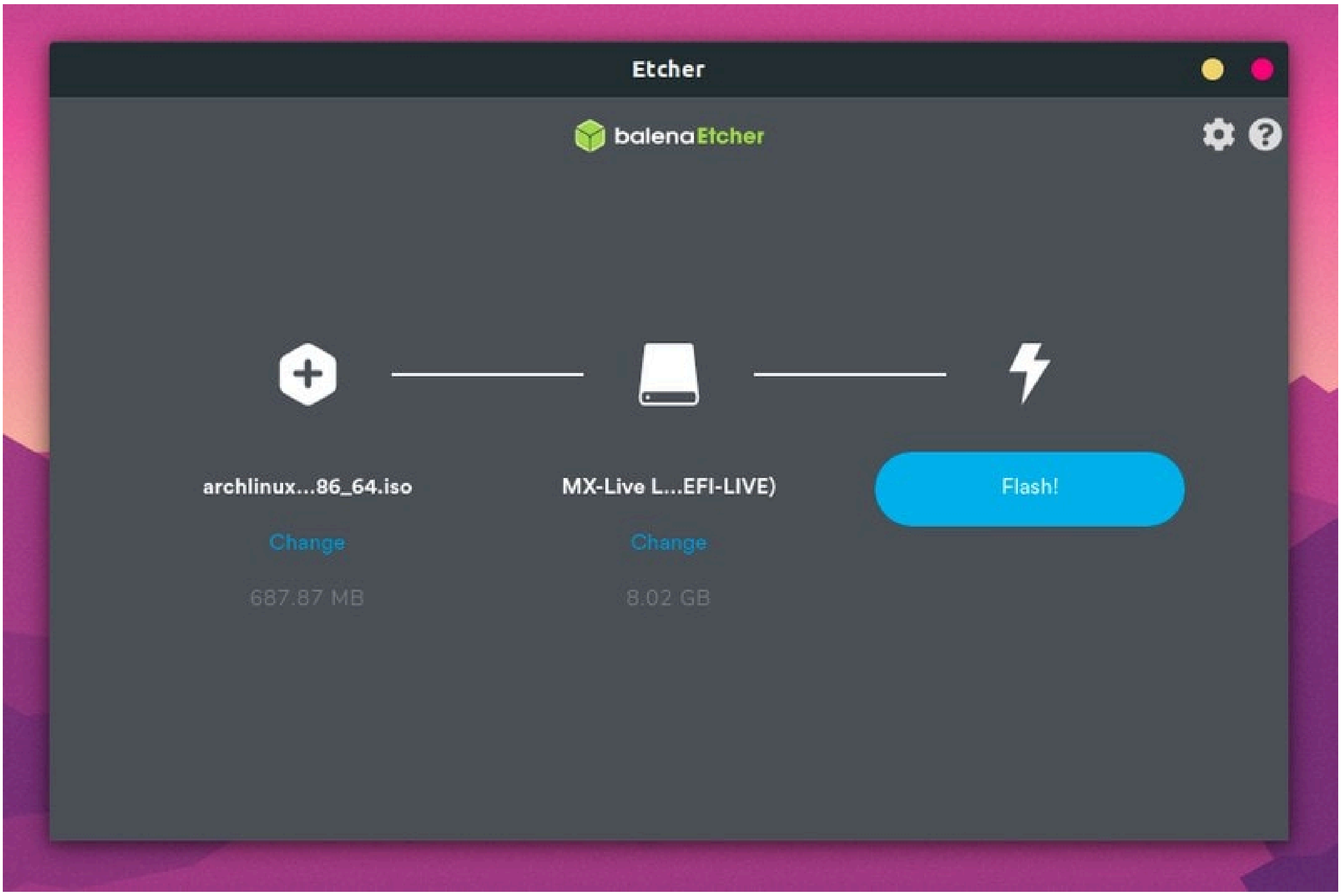
You can download the ISO from the official website. Both direct download and torrent links are available.

Download Arch Linux

Step 2: Create a live USB of Arch Linux

You will have to create a live USB of Arch Linux from the ISO you just downloaded.

You may use the Etcher GUI tool to create the live USB. It is available for both Windows and Linux.



Using Etcher to create Arch Linux live USB

Alternatively, if you are on Linux, you can use the dd command to create a live USB. Replace */path/to/archlinux.iso* with the path where you have downloaded the ISO file, and */dev/sdx* with your USB drive in the example below. You can get your drive information using *lsblk* command.

```
dd bs=4M if=/path/to/archlinux.iso of=/dev/sdx status=progress && sync
```

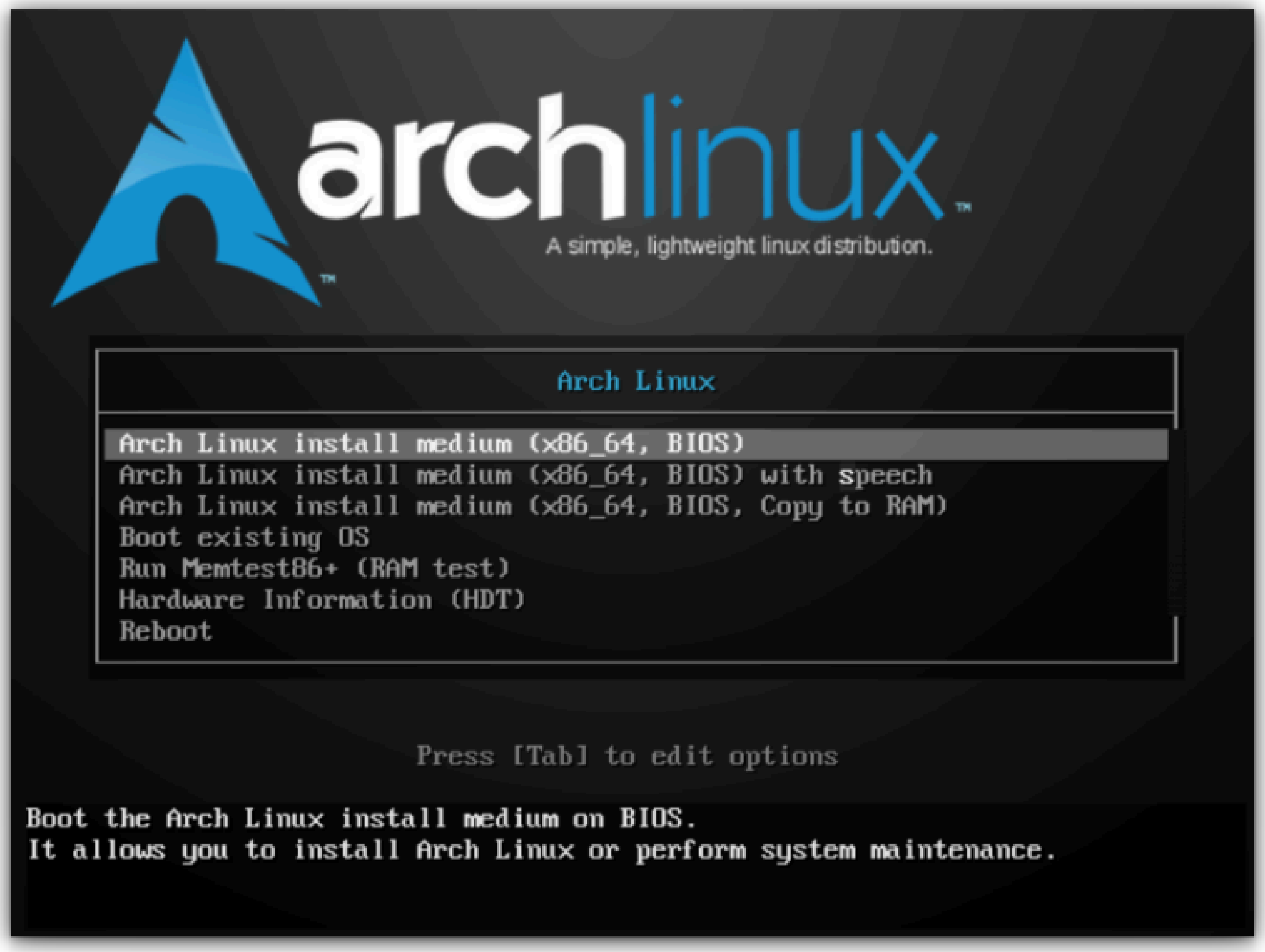
Stick to balenaEtcher if you are comfortable with a GUI.

Step 3: Boot from the live USB

Do note that in some cases, you may not be able to boot from live USB with secure boot enabled. If that's the case with you, disable the secure boot first.

Once you have created a live USB for Arch Linux, shut down your PC. Plugin your USB and boot your system. While booting keep pressing F2, F10 or F12 key (depending upon your system) to go into boot settings.

Here, select to boot from USB or removable disk. Once you do that and the system boots, you should see an option like this:


archlinux
A simple, lightweight linux distribution.
Arch Linux
Arch Linux install medium (x86_64, BIOS)
Arch Linux install medium (x86_64, BIOS) with speech
Arch Linux install medium (x86_64, BIOS, Copy to RAM)
Boot existing OS
Run Memtest86+ (RAM test)
Hardware Information (HDT)
Reboot
Press [Tab] to edit options

```
Boot the Arch Linux install medium on BIOS.
It allows you to install Arch Linux or perform system maintenance.
```

Arch Linux Boot Screen

Select Boot Arch Linux (x86_64). After various checks, Arch Linux will boot to the login prompt with the root user.

Not using US keyboard? Read this

The default keyboard layout in the live session is US. While most English language keyboards will work just fine, the same cannot be true for French, German and other keyboards.

If you face difficulty, you can list out all the supported keyboard layout:

```
ls /usr/share/kbd/keymaps/**/*.map.gz
```

And then change the layout to the an appropriate one using ***loadkeys command***. For example, if you want a German keyboard, this is what you'll use:

```
loadkeys de-latin1
```

The next steps include partitioning the disk, creating the filesystem and mounting it.

📋 ***Again, read all the instructions properly and follow each step carefully. You miss one step or ignore something and you'll have difficulty installing Arch.***

Step 4: Partition the disks

For partitioning the disks, we'll use command line based partition manager fdisk.

Use this command to list all the disk and partitions on your system:

```
fdisk -l
```

📋 **Your hard disk should be labelled /dev/sda or /dev/nvme0n1. Please use the appropriate disk labeling for your system. I am using /dev/sda because that's more common.**

First, select the disk you are going to format and partition:

```
fdisk /dev/sda
```

I suggest that you delete any existing partitions on the disk using command **d**. Once you have the entire disk space free, it’s time to create new partitions with command **n**.

Check if you have UEFI mode enabled

Some steps are different for UEFI and non-UEFI systems.You should verify if you have UEFI enabled system or not. Use this command:

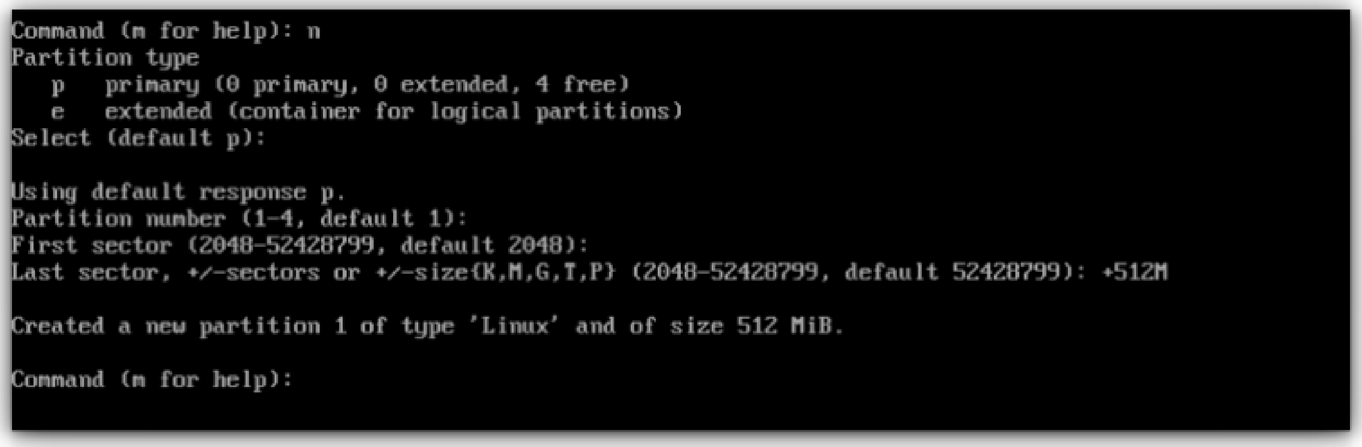
```
ls /sys/firmware/efi/efivars
```

If this directory exists, you have a UEFI enabled system. You should follow the steps for UEFI system. The steps that differ are clearly mentioned.

Create an ESP partition (For UEFI systems only)

If you have a UEFI system, you **must** create an EFI partition at the beginning of your disk. Otherwise, skip this step.

When you enter n, it will ask you to choose a disk number, enter 1. Stay with the default block size, when it asks for the partition size, enter +512M.

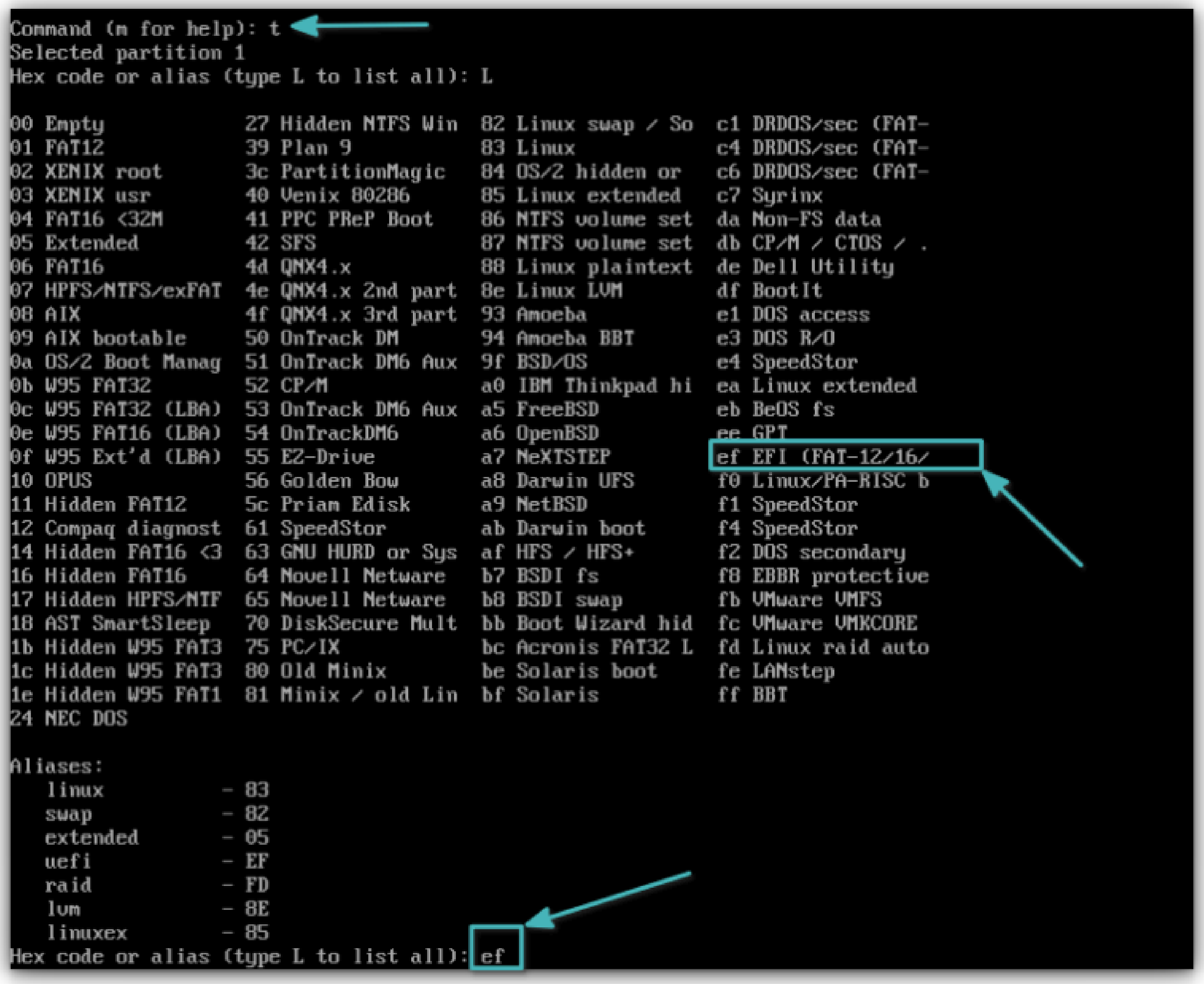


```
Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-52428799, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-52428799, default 52428799): +512M

Created a new partition 1 of type 'Linux' and of size 512 MiB.

Command (m for help):
```

Creating EFI System Partition

One important steps is to change the type of the EFI partition to EFI System (instead of Linux system).

Enter **t** to change type. Enter L to see all the partition types available and then enter its corresponding number to the EFI system.



```
Command (m for help): t
Selected partition 1
Hex code or alias (type L to list all): L

00 Empty            27 Hidden NTFS Win  82 Linux swap / So  c1 DRDOS/sec (FAT-
01 FAT12            39 Plan 9           83 Linux            c4 DRDOS/sec (FAT-
02 XENIX root       3c PartitionMagic   84 OS/2 hidden or   c6 DRDOS/sec (FAT-
03 XENIX usr        40 Venix 80286      85 Linux extended   c7 Syrinx
04 FAT16 <32M       41 PPC PReP Boot    86 NTFS volume set  da Non-FS data
05 Extended         42 SFS              87 NTFS volume set  db CP/M / CTOS / .
06 FAT16            4d QNX4.x           88 Linux plaintext  de Dell Utility
07 HPFS/NTFS/exFAT  4e QNX4.x 2nd part  8e Linux LVM        df BootIt
08 AIX              4f QNX4.x 3rd part  93 Amoeba           e1 DOS access
09 AIX bootable     50 OnTrack DM       94 Amoeba BBT       e3 DOS R/O
0a OS/2 Boot Manag  51 OnTrack DM6 Aux  9f BSD/OS           e4 SpeedStor
0b W95 FAT32        52 CP/M             a0 IBM Thinkpad hi  ea Linux extended
0c W95 FAT32 (LBA)  53 OnTrack DM6 Aux  a5 FreeBSD          eb BeOS fs
0e W95 FAT16 (LBA)  54 OnTrackDM6       a6 OpenBSD          ee GPT
0f W95 Ext'd (LBA)  55 EZ-Drive         a7 NeXTSTEP         ef EFI (FAT-12/16/
10 OPUS             56 Golden Bow       a8 Darwin UFS       f0 Linux/PA-RISC b
11 Hidden FAT12     5c Priam Edisk      a9 NetBSD           f1 SpeedStor
12 Compaq diagnost  61 SpeedStor        ab Darwin boot      f4 SpeedStor
14 Hidden FAT16 <3  63 GNU HURD or Sys  af HFS / HFS+       f2 DOS secondary
16 Hidden FAT16     64 Novell Netware   b7 BSDI fs          f8 EBBR protective
17 Hidden HPFS/NTF  65 Novell Netware   b8 BSDI swap        fb VMware VMFS
18 AST SmartSleep   70 DiskSecure Mult  bb Boot Wizard hid  fc VMware VMKCORE
1b Hidden W95 FAT3  75 PC/IX            bc Acronis FAT32 L  fd Linux raid auto
1c Hidden W95 FAT3  80 Old Minix        be Solaris boot     fe LANstep
1e Hidden W95 FAT1  81 Minix / old Lin  bf Solaris          ff BBT
24 NEC DOS

Aliases:
   linux          - 83
   swap           - 82
   extended       - 05
   uefi           - EF
   raid           - FD
   lvm            - 8E
   linuxex        - 85
Hex code or alias (type L to list all): ef
```

Change Type of EFI System Partition

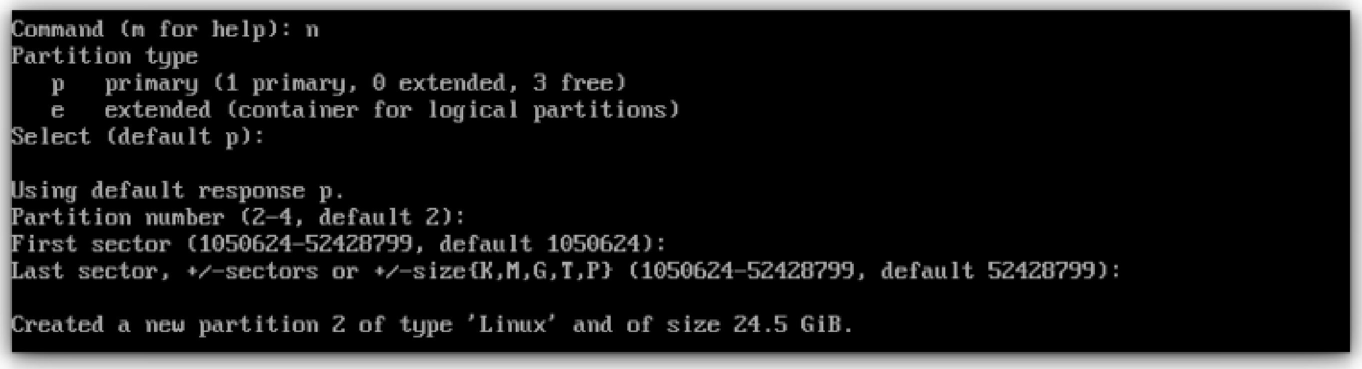
Create root partition

You need to create root partition ***for both UEFI and legacy systems***.

The common partitioning practice was/is to create root, swap and home partitions separately. You may just create a single root partition and create a swapfile and home under the root directory itself.

So, in this approach, we'll have a single root partition, no swap, no home.

While you are in the fdisk command, press n to create a new partition. It will automatically give it partition number 2. This time keep on pressing enter to allocate entire remaining disk space to the root partition.



```
Command (m for help): n
Partition type
   p   primary (1 primary, 0 extended, 3 free)
   e   extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (2-4, default 2):
First sector (1050624-52428799, default 1050624):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (1050624-52428799, default 52428799):

Created a new partition 2 of type 'Linux' and of size 24.5 GiB.
```

New Partition

When you are done with the disk partitioning, enter **w** command to write the changes to the disk and exit out of fdisk command.

Step 4: Create filesystem

Now that you have your disk partitions ready, it’s time to create filesystem on it. Follow the steps for your system

Creating filesystem for UEFI system

So, you have two disk partitions and the first one is EFI type. Create a FAT32 file system on it using the mkfs command:

```
mkfs.fat -F32 /dev/sda1
```

Now create an Ext4 filesystem on the root partition:

```
mkfs.ext4 /dev/sda2
```

Creating filesystem for non-UEFI system

For non-UEFI system, you only have one single root partition. So just make it ext4:

```
mkfs.ext4 /dev/sda1
```

Step 5: Connect to WiFi

You can connect to WiFi interactively using this helpful utility called iwctl. Just enter this command and follow the on-screen instructions:

```
iwctl
```

Next, you can list all your wireless interfaces/devices connected using the command:

```
device list
```

You need to select the preferred one.

Once you select the wireless interface, scan for available network using the command below:

```
station wlan0 scan
```

While it scans for the network, you don’t get to see the network names yet. So, to see the connections available, you can type in:

```
station wlan0 get-networks
```

Among the listed networks, you can connect to your target Wi-Fi using the command:

```
station wlan0 connect "Name of Network/WiFi"
```

If it is protected by a password, you will be asked for it, enter the credentials and you should be connected to it.

Exit the network setup prompt using **Ctrl + D**.

Now, we're connected to the network, but to make sure, you can check if you could use the internet by using the ping command:

```
ping google.com
```

If you get bytes in reply, you are connected. Use **Ctrl+C** to stop the ping reply.

Step 6: Select an appropriate mirror

This is a big problem with installing Arch Linux. If you just go on installing it, you might find that the downloads are way too slow. In some cases, it's so slow that the download fails.

It's because the mirrorlist (located in /etc/pacman.d/mirrorlist) has a huge number of mirrors but not in a good order. The top mirror is chosen automatically and it may not always be a good choice.

Thankfully, there is a fix for that. First sync the pacman repository so that you can download and install software:

```
pacman -Syy
```

Now, install reflector too that you can use to list the fresh and fast mirrors located in your country:

```
pacman -S reflector
```

Make a backup of mirror list (just in case):

```
cp /etc/pacman.d/mirrorlist /etc/pacman.d/mirrorlist.bak
```

Now, get the good mirror list with reflector and save it to mirrorlist. You can change the country from US to your own country.

```
reflector -c "US" -f 12 -l 10 -n 12 --save /etc/pacman.d/mirrorlist
```

All good to go now.

Step 7: Install Arch Linux

Since you have all the things ready, it's time to finally install the Arch Linux. You'll be installing it on the root directory so mount it first.

Do you remember the name of the root partition? Use it to mount it:

```
mount /dev/sda2 /mnt
```

Note that this is valid for UEFI systems, you will have to use **/dev/sda1** in non-UEFI systems.

With root mounted, it's time to use the wonderful pacstrap script to install all the necessary packages:

```
pacstrap /mnt base linux linux-firmware vim nano
```

It will take some time to download and install these packages. If the downloads get interrupted, no need to panic. You can run the above command once again and it resumed the download.

I have added Vim and Nano text editor to the list because you'll need to edit some files post-installation. You can use anything you are comfortable with.

For reference, head to our nano vs vim article to explore the differences between these editors.

Step 8: Configure the installed Arch system

Generate a fstab file to define how disk partitions, block devices, or remote file systems are mounted into the filesystem.

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

Now use arch-chroot and enter the mounted disk as root. Actually, now you are using the just installed Arch Linux system on the disk. You'll have to do some configuration changes to the installed system so that you could run it properly when you boot from the disk.

```
arch-chroot /mnt
```

Setting Timezone

To set up timezone on Linux, you can use timedatectl command. First find your time zone:

```
timedatectl list-timezones
```

Exit from the list using **Ctrl + C** or just **q**. And then set it up like this (replace Europe/Paris with your desired time zone):

```
timedatectl set-timezone Europe/Paris
```

Setting up Locale

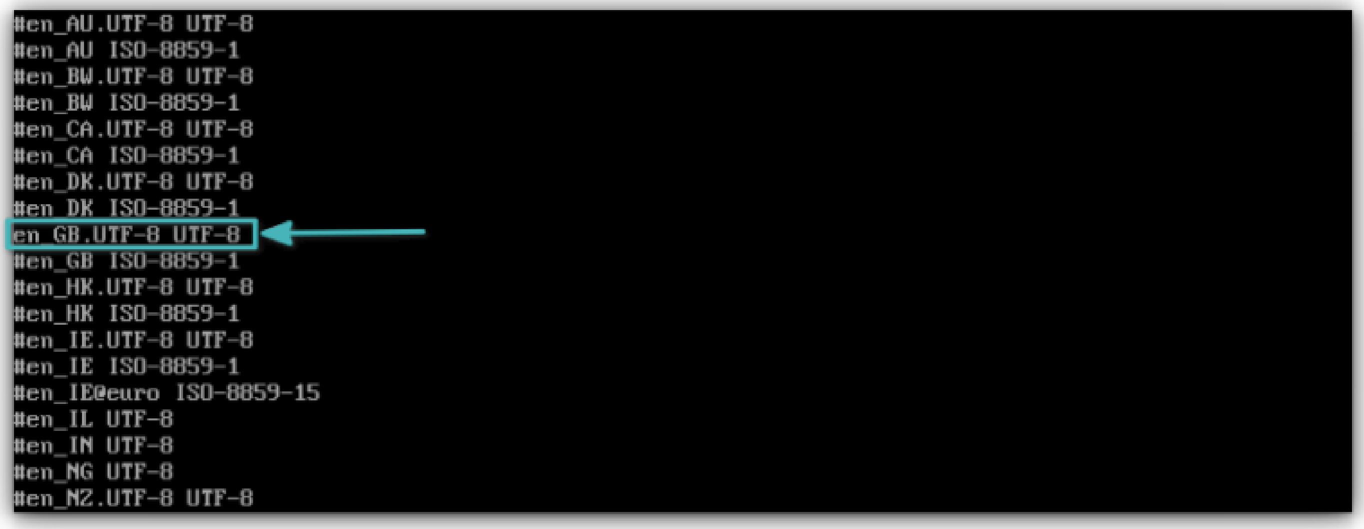
This is what sets the language, numbering, date, and currency formats for your system.

The file ***/etc/locale.gen*** contains all the local settings and system language in a commented format.

Open the file using Vim or Nano editor and uncomment (remove the # from the start of the line) the language you prefer.

The command to open the file looks like:

```
nano /etc/locale.gen
```



```
#en_AU.UTF-8 UTF-8
#en_AU ISO-8859-1
#en_BW.UTF-8 UTF-8
#en_BW ISO-8859-1
#en_CA.UTF-8 UTF-8
#en_CA ISO-8859-1
#en_DK.UTF-8 UTF-8
#en_DK ISO-8859-1
en_GB.UTF-8 UTF-8
#en_GB ISO-8859-1
#en_HK.UTF-8 UTF-8
#en_HK ISO-8859-1
#en_IE.UTF-8 UTF-8
#en_IE ISO-8859-1
#en_IE@euro ISO-8859-15
#en_IL UTF-8
#en_IN UTF-8
#en_NG UTF-8
#en_NZ.UTF-8 UTF-8
```

Locale Gen

I have used **en_GB.UTF-8** (English with Great Britain). Hit **Ctrl + X** and then **Y** to save the selection and continue.

Now generate the locale config in the **/etc** directory file using the below commands one by one:

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

```
export LANG=en_GB.UTF-8
```

Both locale and timezone settings can be changed later on as well when you are using your Arch Linux system.

Network configuration

Create a ***/etc/hostname*** file and add the hostname entry to this file. Hostname is basically the name of your computer on the network.

In my case, I'll set the hostname as ***myarch***. You can choose whatever you want:

```
echo myarch > /etc/hostname
```

The next part is to create the hosts file:

```
touch /etc/hosts
```

And edit this /etc/hosts file with Vim or Nano editor to add the following lines to it (replace myarch with hostname you chose earlier):

```
127.0.0.1   localhost
::1     localhost
127.0.1.1   myarch
```

Set up root password

You should also set the password for the root account using the **passwd** command:

```
passwd
```

You will be prompted to enter (and re-type) your password to confirm.

Step 9: Install Grub bootloader

This is one of the crucial steps and it differs for UEFI and non-UEFI systems. Let me mention the **steps for the UEFI systems** first.

Make sure that you are still using **arch-chroot**. Install required packages:

```
pacman -S grub efibootmgr
```

Create the directory where EFI partition will be mounted:

```
mkdir /boot/efi
```

Now, mount the ESP partition you had created

```
mount /dev/sda1 /boot/efi
```

Install grub like this:

```
grub-install --target=x86_64-efi --bootloader-id=GRUB --efi-directory=/boot/efi
```

One last step:

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

Install grub on Non-UEFI systems

Install grub package first:

```
pacman -S grub
```

And then install grub like this (don’t put the disk number sda1, just the disk name sda):

```
grub-install /dev/sda
```

Last step:

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

Step 10: Create Additional user and enforce privileges

You should not boot into a system, which has only a root user account. This way, every change you make will happen without any authentication required, and you might end up messing up your system.

Of course, you can still choose to do it, but it is not the recommended solution for a stable and secure experience.

So there should be an additional user, who gets root privileges using sudo. That reminds me, you should also install the sudo package if you didn’t already:

```
pacman -S sudo
```

Now create a new user and give permissions. In my case, ‘team’ is the new username I chose. You can select your own.

```
useradd -m team
passwd team
```

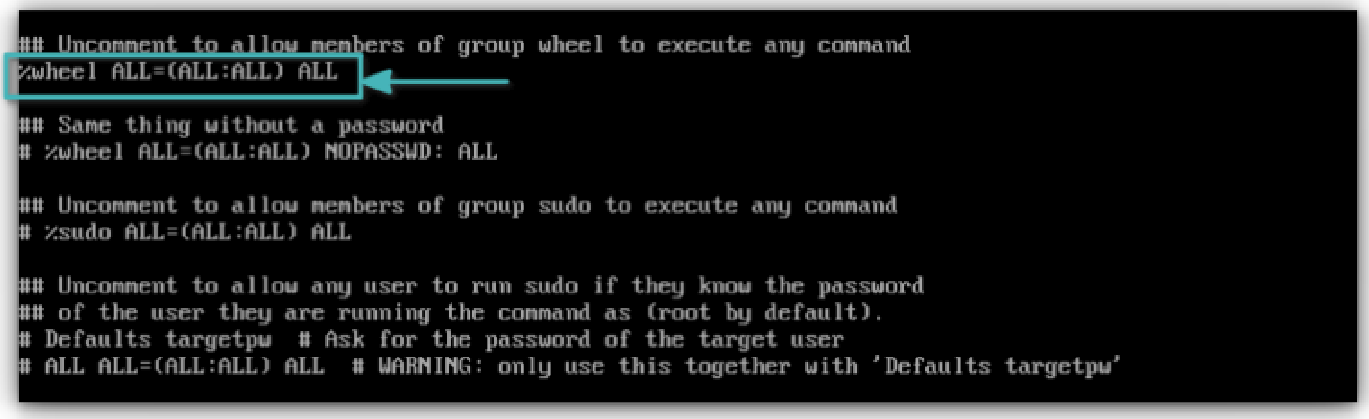
Enter the password for this user and confirm. Now, you will be adding this user to a group of users that grants specific permissions. This should be self-explanatory, while the wheel group is needed for a user act as the super user.

```
usermod -aG wheel,audio,video,storage team
```

Finally, you need to edit the visudo file, specifically the line referring to wheel should be uncommented as shown in the image below.

It opens in VI editor by default. So we need to force it use nano:

```
EDITOR=nano visudo
```



```
## Uncomment to allow members of group wheel to execute any command
%wheel ALL=(ALL:ALL) ALL

## Same thing without a password
# %wheel ALL=(ALL:ALL) NOPASSWD: ALL

## Uncomment to allow members of group sudo to execute any command
# %sudo ALL=(ALL:ALL) ALL

## Uncomment to allow any user to run sudo if they know the password
## of the user they are running the command as (root by default).
# Defaults targetpw  # Ask for the password of the target user
# ALL ALL=(ALL:ALL) ALL  # WARNING: only use this together with 'Defaults targetpw'
```

Edit visudo

Save the changes and exit from this file.

Step 11: Install a desktop environment (GNOME in this case)

The first step is to install the X environment. Type the below command to install the Xorg as display server along with the network manager. You can refer to the official documentation for Wayland.

```
pacman -S xorg networkmanager
```

Now, you can install GNOME desktop environment on Arch Linux using:

```
pacman -S gnome
```

> 📋 This will give you a huge list of packages. If you want to handpick packages to install, use the corresponding number. You can also use ranges to select a continuous group of packages. For example, 1,2,4-8,9 installs the first, second, fourth to eighth, and ninth packages, skipping the third. Be mindful of what you're skipping.

The last step includes enabling the display manager GDM for Arch. I also suggest enabling Network Manager

```
systemctl enable gdm.service
systemctl enable NetworkManager.service
```

Now exit from chroot using the exit command:

```
exit
```

Finally, unmount the root partition using the following command:

```
umount /mnt
```

Or,

```
umount -l /mnt
```

And then shut down your system

```
shutdown now
```

Don’t forget to take out the live USB before powering on the system again. If everything goes well, you should see the Grub screen and then the GNOME login screen.

If you want a KDE desktop, please follow this tutorial about installing KDE on Arch Linux.

Final Words on Arch Linux installation

You might have realized by now that installing Arch Linux is not as easy as installing Ubuntu. However, with a little patience, you can surely accomplish it and then tell the world that you use Arch Linux.

Arch Linux installation itself provides a great deal of learning. I recommend a few essential things to do after installing Arch Linux where you’ll find steps to install various other desktop environments (if you didn’t install it here) and learn more about the OS.

Things to do after Installing Arch Linux
TO IMPROVE YOUR ARCH EXPERIENCE
IT'S FOSS

7 Essential Things To Do After Installing Arch Linux

Brief: This tutorial shows you a few essential things to do after installing Arch Linux. This will help you get started with Arch Linux so that you can explore it further. Earlier I showed you how to install Ar…

It's FOSS • Abhishek Prakash

You can keep playing with it and see how powerful Arch is.

Let us know in the comments if you face any difficulty while installing Arch Linux.

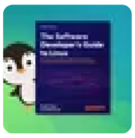
Arch Linux Tutorial

ABOUT THE AUTHOR

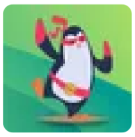
Abhishek Prakash

Created It's FOSS 11 years ago to share my Linux adventures. Have a Master's degree in Engineering and years of IT industry experience. Huge fan of Agatha Christie detective mysteries 🕵️‍♀️

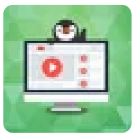
Featured



Teaching Linux to Software Developers With This Book



7 Sudo Tips and Tweaks for Linux Users



11 YouTube Channels Linux Users Should Explore

Latest

FOSS Weekly #24.20: Linux Kernel 6.9, SnowflakeOS, Nvidia GPU Handling and More

16 May 2024

Switching Between Intel and Nvidia Graphics Cards on Ubuntu

15 May 2024

Testing the Surfshark VPN GUI App on Linux

14 May 2024

Become a Better Linux User

With the FOSS Weekly Newsletter, you learn useful Linux tips, discover applications, explore new distros and stay updated with the latest from Linux world

Your email address

Subscribe

What's your reaction?

🔥 34	😍 8	😮 3	😢 2	😂 5	😠 3
superb	love	wow	sad	laugh	angry

337 Comments 19 ONLINE LOGIN

We welcome and encourage civil, courteous, and community-friendly comments.

Write your comment...

Newest ⌄

James 1 year ago FEATURED

I'm very new to Linux and only knew the cd command before now, and yet with this guide I installed supposedly one of the hardest distributions first try. Thank you for being so thorough, and thank you for writing this guide!

👍 0 👎 0 Reply

Abhishek Admin 1 year ago

Wow! That's an amazing feat, James :)

👍 0 👎 0 Reply

Guest 1 month ago

Please be aware that a YouTuber is stealing your content and publishing it as their own: https://boisterous-toffee-60cc83.netlify.app/how-to-use-ventoy-to-create-a-multiboot-usb/

👍 0 👎 0 Reply

space_cowboy 4 months ago

Thanks for all your amazing content! For what it's worth, looks like some systemd tools don't work at the stage you mentioned them (I eventually found it on the wiki). I ended up using `tzselect` because of this. I can't imagine you would include steps that don't work so maybe you could elaborate on why what you wrote should work??? So far, so good with everything else. Much appreciation.

👍 0 👎 0 Reply

Abhishek Plus Member ⭐ 4 months ago

You mean enabling gdm and network manager? I don't see why it would not work. What error did you see (if you can remember it)?

👍 0 👎 0 Reply

space_cowboy 4 months ago

The error occurred when I attempted to run `timedatectl` . It was an error associated with systemd 'not running', something along those lines.

After some searching, I found on the Arch wiki (linked in my original comment - that's actually the wrong link, sorry about that, I mean here: "...during chroot... `timedatectl` will not work...").

There's also "Use systemd-firstboot(1) to initialize the system time zone for mounted (but not booted) system images." mentioned here.
But I still feel like I may be missing something in my understanding...

👍 0 👎 0 Reply

Head_on_a_Stick 4 months ago

Users shouldn't be added to the video, audio or storage groups because systemd-logind takes care of device access. The wheel group is not needed for superuser access, that is only true for the BSDs.

👍 0 👎 0 Reply

pigmanbruh 6 months ago

i think you forgot to add about parallel download in pacman.conf, the speed difference absolutely slaps

👍 0 👎 0 Reply

Abhishek Plus Member ★ 6 months ago

Wanted to keep it for 'things to do after installing Arch'

👍 0 👎 0 Reply

Load More Comments

READ NEXT

How to Install VirtualBox on Ubuntu [Beginner's Tutorial] ›

Running AI Locally Using Ollama on Ubuntu Linux ›

How to Use Pop Shell on GNOME Desktop ›

Enabling Bluetooth on Arch Linux ›

Fixing 'target not found' Error in Arch Linux ›

Navigation

- 📰 News
- 📬 Newsletter
- 🧩 Quizzes & Puzzles
- 🎒 Resources
- 🏘️ Community
- 📖 About
- 🪪 Contact
- 📜 Policies
- Linux Server Side
- En Español
- 🗳️ Feedback

Resources

Distro Resources 📖

Guides 📒

Courses 🎓

Social

Facebook

Twitter

RSS

Instagram

Telegram

Youtube

○ System ⇕