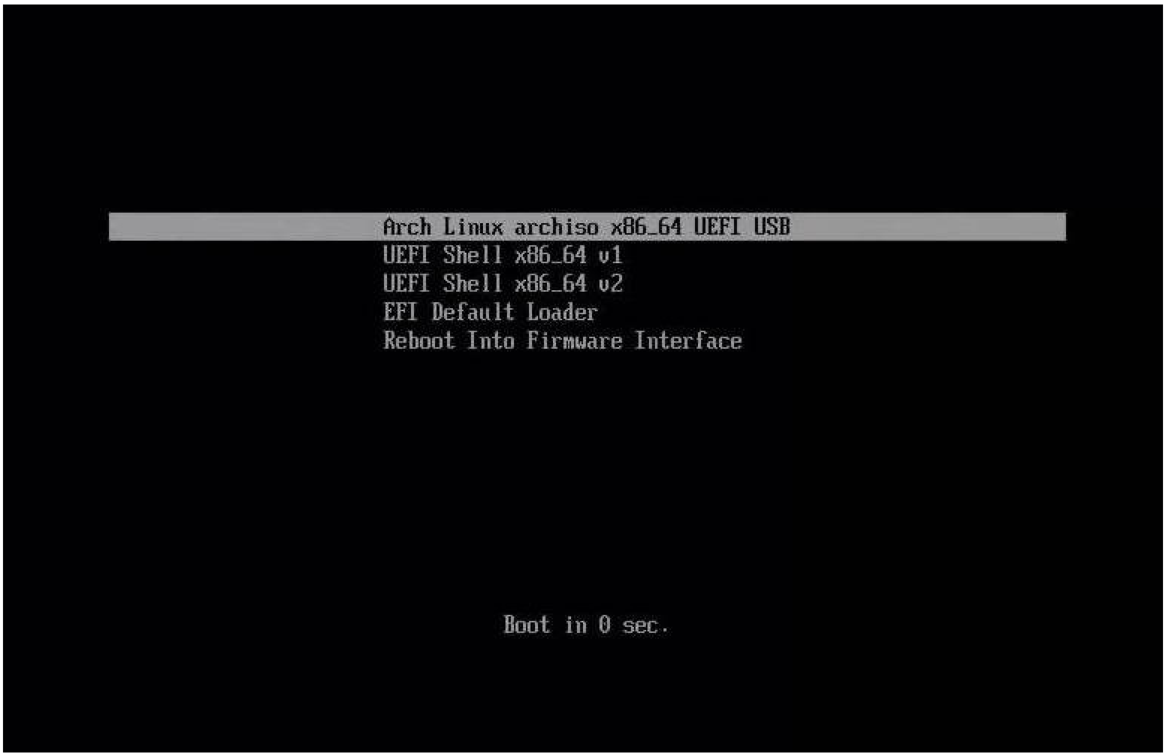
A step by step Arch Linux installation guide

alu



```
Arch Linux archiso x86_64 UEFI USB
UEFI Shell x86_64 v1
UEFI Shell x86_64 v2
EFI Default Loader
Reboot Into Firmware Interface

Boot in 0 sec.
```

Arch Linux Installation live boot

Many Linux users are stopped from using Arch Linux because they cannot install it. This Arch Linux installation guide will solve this problem. It shows the whole Arch Linux installation process step by step.

Summary

UEFI or legacy mode?

There are two possible options of Arch Linux installation:

- legacy mode
- UEFI mode

I personally think that the Arch Linux installation in the legacy mode is more **conservative, simpler, and**

reliable. However, the UEFI mode is more **up-to-date and some modern hardware** supports only the UEFI installation.

This Arch Linux installation guide will use the **UEFI mode**, but I will point out the steps and commands that are different for the **legacy mode**. So, regardless of what mode you choose, this guide will help you to install Arch Linux with a minimal graphical environment.

The UEFI means Unified Extensible Firmware Interface and here are some of its benefits:

- UEFI replaces the Basic Input/Output System (BIOS) interface.
- UEFI is compatible with GPT tables.
- It supports larger hard drives.
- It is more configurable and, as a result, it boots faster.

Also, I will use a GUID Partition Table (GPT). It is more advanced than a master boot record (MBR) partitions. It works with volumes larger than 2TB and supports up to 128 partitions. However, if you decide to do the legacy installation, use the MBR table.

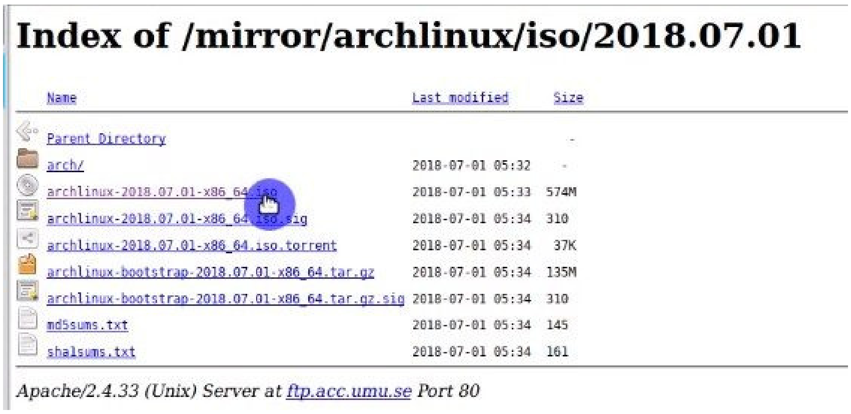
So, let's get started with the detailed Arch Linux UEFI installation guide.

Before you start

Download Arch Linux ISO

First, download the Arch Linux installation ISO from the Arch Linux website.

You can download the torrent file or use the direct link. Just find your country and click on the link. Among the variety of files, choose the ISO `archlinux-xxxx.xx.xx-x86_64.iso` and signature `archlinux-xxxx.xx.xx-x86_64.iso.sig` files:



Index of /mirror/archlinux/iso/2018.07.01

Name	Last modified	Size
Parent Directory		-
arch/	2018-07-01 05:32	-
archlinux-2018.07.01-x86_64.iso	2018-07-01 05:33	574M
archlinux-2018.07.01-x86_64.iso.sig	2018-07-01 05:34	310
archlinux-2018.07.01-x86_64.iso.torrent	2018-07-01 05:34	37K
archlinux-bootstrap-2018.07.01-x86_64.tar.gz	2018-07-01 05:34	135M
archlinux-bootstrap-2018.07.01-x86_64.tar.gz.sig	2018-07-01 05:34	310
md5sums.txt	2018-07-01 05:34	145
sha1sums.txt	2018-07-01 05:34	161

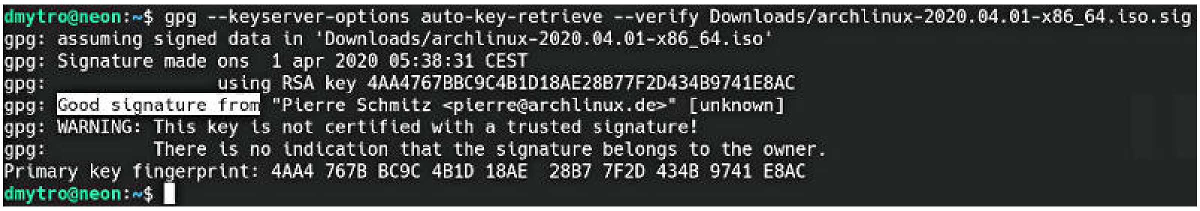
Apache/2.4.33 (Unix) Server at ftp.acc.umu.se Port 80

Download the Arch Linux ISO

When the ISO is downloaded, you need to check its signature to make sure it has not been compromised:

```
gpg --keyserver-options auto-key-retrieve --verify /path/to/archlinux.iso.sig
```

If you see *"Good signature from …"*, this means everything is alright:



```
dmytro@neon:~$ gpg --keyserver-options auto-key-retrieve --verify Downloads/archlinux-2020.04.01-x86_64.iso.sig
gpg: assuming signed data in 'Downloads/archlinux-2020.04.01-x86_64.iso'
gpg: Signature made ons  1 apr 2020 05:38:31 CEST
gpg:                using RSA key 4AA4767BBC9C4B1D18AE28B77F2D434B9741E8AC
gpg: Good signature from "Pierre Schmitz <pierre@archlinux.de>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 4AA4 767B BC9C 4B1D 18AE  28B7 7F2D 434B 9741 E8AC
dmytro@neon:~$ 
```

Next, you need to write it to your USB flash drive. Open the Linux terminal and use the following command:

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

where `/path/to/archlinux.iso` is the path to your downloaded ISO file; `/dev/sdx` is your flash drive. You can find out this name with the command `sudo fdisk -l` and the size of your USB flash drive.

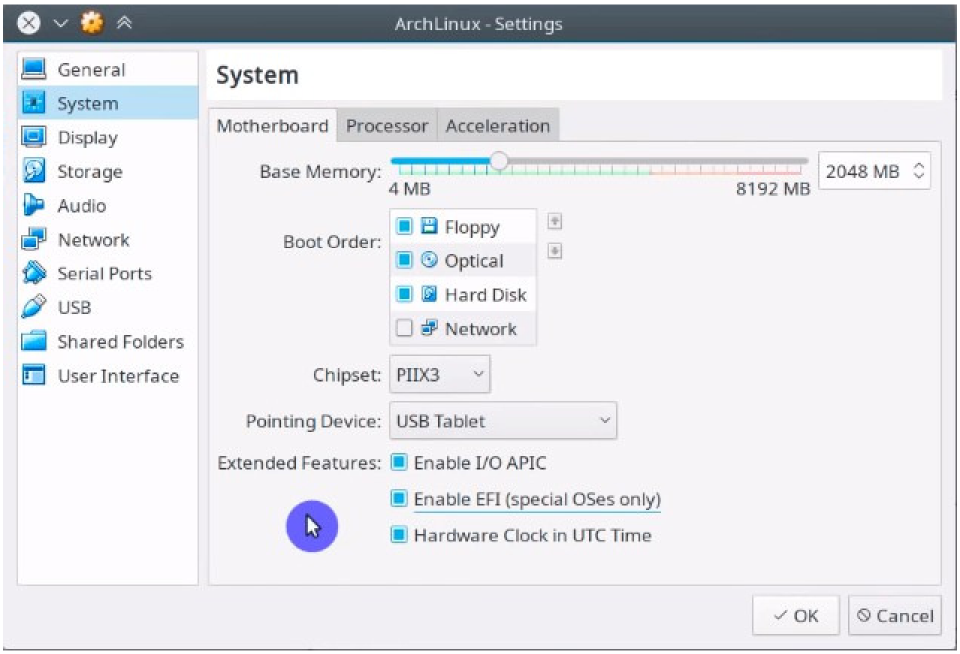
Alternatively, you can use a graphical program to write ISO to a USB.

Practice your Arch Linux installation in VirtualBox

You may try the Arch Linux on a virtual machine first. It will help you to experience the installation and avoid errors in the future real installations. To run the Arch Linux in VirtualBox, do the following steps.

1. Create a new machine and name it Arch Linux.
2. Give it 2GB of RAM. If you lack the RAM on your PC, keep it at 1GGB.
3. Create a new hard disk. Select VDI (VirtualBox Disk Image) and dynamically allocated size.

4. Give the VDI 20GB and click Create.
5. Click on the Settings icon. In the System tab, enable the UEFI mode.
6. Click on the Start icon or just double-click on the Arch Linux.
7. Specify the path to the downloaded Arch Linux ISO.



ArchLinux - Settings

System

Base Memory: 2048 MB (4 MB – 8192 MB)

Boot Order:
☑ Floppy
☑ Optical
☑ Hard Disk
☐ Network

Chipset: PIIX3

Pointing Device: USB Tablet

Extended Features:
☑ Enable I/O APIC
☑ Enable EFI (special OSes only)
☑ Hardware Clock in UTC Time

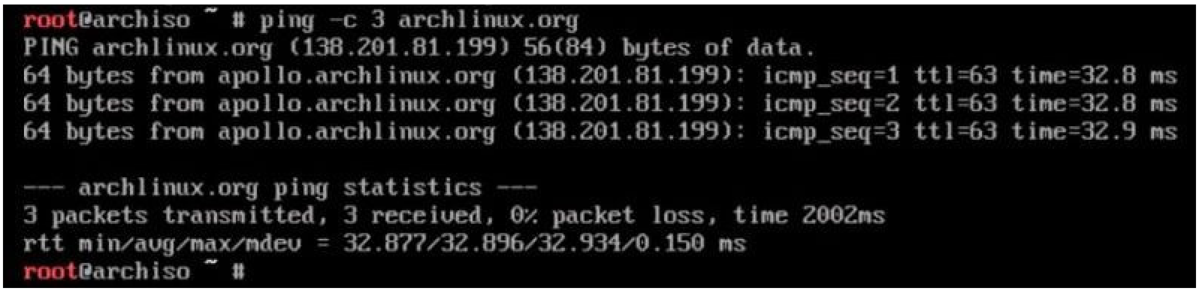
VirtualBox in the EFI mode

The boot process of live Arch Linux in the UEFI mode can be very slow. If you just see the back screen, this is normal. Just wait some time. When the system is loaded, and you can start the Arch Linux installation.

Arch Linux installation

Check network connection

First of all, check the internet connection. I recommend you to use a wired connection. To check if your internet works, you need to ping to any server, for example, the Arch Linux website:



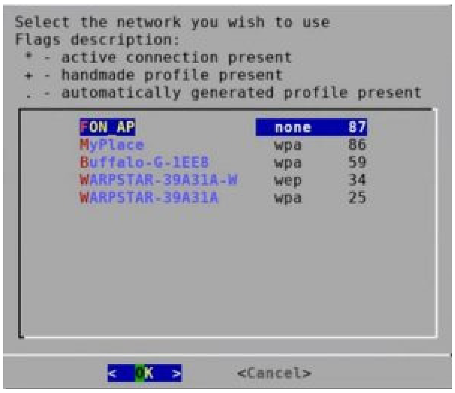
```
root@archiso ~ # ping -c 3 archlinux.org
PING archlinux.org (138.201.81.199) 56(84) bytes of data.
64 bytes from apollo.archlinux.org (138.201.81.199): icmp_seq=1 ttl=63 time=32.8 ms
64 bytes from apollo.archlinux.org (138.201.81.199): icmp_seq=2 ttl=63 time=32.8 ms
64 bytes from apollo.archlinux.org (138.201.81.199): icmp_seq=3 ttl=63 time=32.9 ms

--- archlinux.org ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 32.877/32.896/32.934/0.150 ms
root@archiso ~ #
```

Test Arch Linux connection with ping

If you are not sure what interfaces are available, use

If you use a wired connection, it is usually picked up automatically. Wi-Fi requires some additional settings. For Wi-Fi run:

You will see a window looking like this where you can choose the available networks:



```
Select the network you wish to use
Flags description:
 * - active connection present
 + - handmade profile present
 . - automatically generated profile present

        FON AP                  none    87
        MyPlace                 wpa     86
        Buffalo-G-1EE8          wpa     59
        WARPSTAR-39A31A-W       wep     34
        WARPSTAR-39A31A         wpa     25

               <  OK  >         <Cancel>
```

WI-Fi menu in Arch Linux

Type the password and connect to your Wi-Fi network. However, I recommend using a wired internet connection when possible.

Partition

The next step in our Arch Linux installation guide is to partition the hard drive.

List the available partitions and disks:

Most probably, you will have only two hard drives:

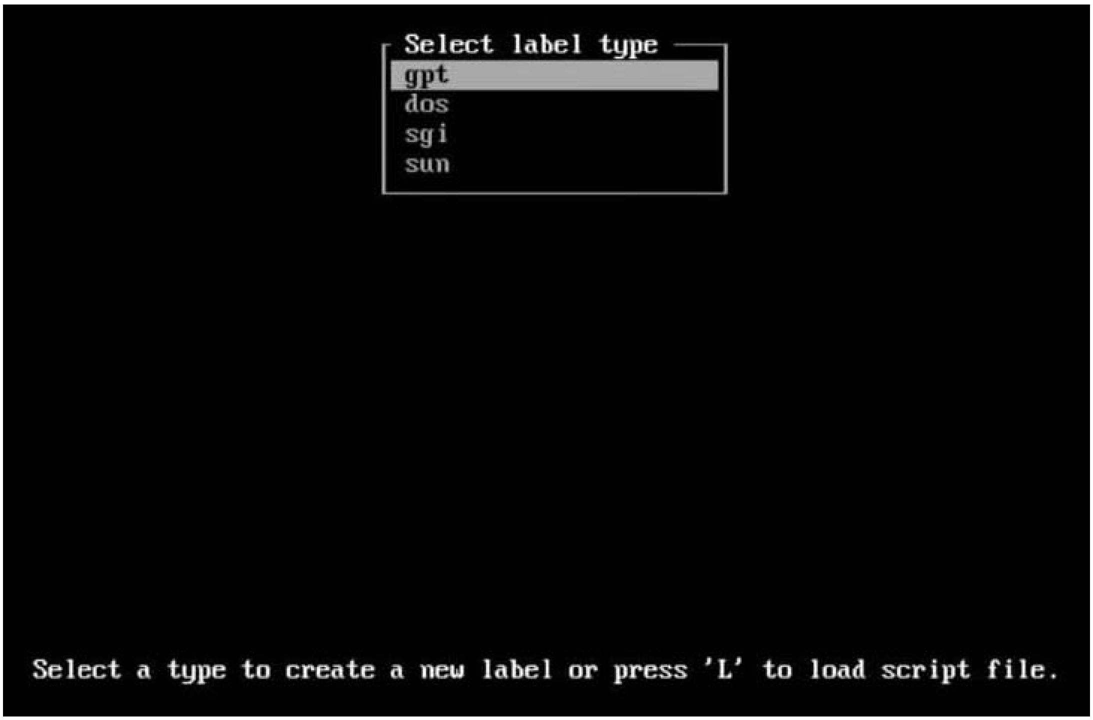
- the USB drive with the Arch Linux ISO
- your computer HDD/SSD.

If you have several hard drives, use `fdisk` to look at their size and define which one you want to use for the Arch Linux installation.

If you already have a partition table, skip this step. If your hard drive is brand-new as in the case of a virtual machine or you want to re-partition your hard drive, run this command to create a new partition table:

Note! *Back up all your data, because creating a new partition table will erase everything from a drive.*

In the label type window, select *GPT*.



```
Select label type
gpt
dos
sgi
sun

Select a type to create a new label or press 'L' to load script file.
```

Select GPT partition

Legacy mode! *If you do the legacy installation, choose the `dos` partition type and do not create the UEFI partition.*

Use the arrow keys and `Enter` to create 3 partitions with `cfdisk`:

- /dev/sda1 # choose 512Mb of space (UEFI)
- /dev/sda2 # choose at least 10 GB of space (root)
- /dev/sda3 # choose all the left space (home)

Write the table to your hard drive and quit.

If you are not familiar with `cfdisk`, watch the video tutorial above.

Now, list the partitions again:

The `/dev/sda` disk should have three partitions. We need to format them.

The first partition is the UEFI partition. It needs to be formatted with a FAT file system:

Legacy mode! *If you do the legacy installation, skip the step of UEFI formatting because you should not have the UEFI partition.*

The other two partitions can be formatted in any Linux file system. I recommend using EXT4:

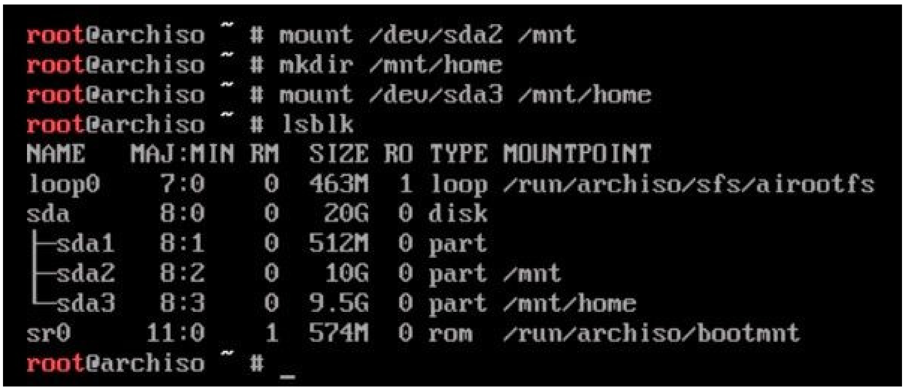
```
mkfs.ext4 /dev/sda2
mkfs.ext4 /dev/sda3
```

Next, mount the root partition:

Create a folder to mount the home partition and mount it:

```
mkdir /mnt/home
mount /dev/sda3 /mnt/home
```

Check mounting points whether they were created successfully:



```
root@archiso ~ # mount /dev/sda2 /mnt
root@archiso ~ # mkdir /mnt/home
root@archiso ~ # mount /dev/sda3 /mnt/home
root@archiso ~ # lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0  463M  1 loop /run/archiso/sfs/airootfs
sda      8:0    0   20G  0 disk
├─sda1   8:1    0  512M  0 part
├─sda2   8:2    0   10G  0 part /mnt
└─sda3   8:3    0  9.5G  0 part /mnt/home
sr0     11:0    1  574M  0 rom  /run/archiso/bootmnt
root@archiso ~ # _
```

Note, the root mounting point is the folder where the system will be installed.

Install the system

Now, we start the installation process by installing the minimal Arch Linux system:

```
pacstrap -i /mnt base linux linux-firmware sudo nano
```

When the system requests to choose the components to install, select *all* and *yes*. Wait some time until it completes.

Generate fstab file

Next step in this Arch Linux installation guide is to generate the *fstab* file:

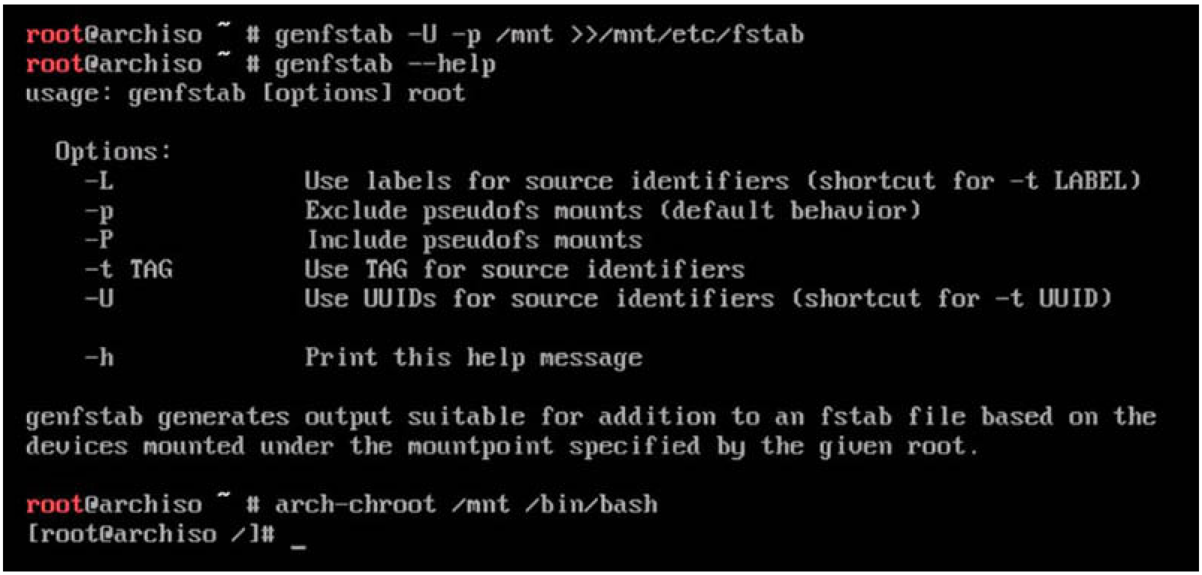
```
genfstab -U -p /mnt >> /mnt/etc/fstab
```

To learn what `-U` and `-p` mean, type `genfstab -help` to see the description of these options.

Chroot to the installed system

Next, chroot (change root) to your system that is mounted to /mnt using the BASH environment:

```
arch-chroot /mnt /bin/bash
```



```
root@archiso ~ # genfstab -U -p /mnt >>/mnt/etc/fstab
root@archiso ~ # genfstab --help
usage: genfstab [options] root

  Options:
    -L            Use labels for source identifiers (shortcut for -t LABEL)
    -p            Exclude pseudofs mounts (default behavior)
    -P            Include pseudofs mounts
    -t TAG        Use TAG for source identifiers
    -U            Use UUIDs for source identifiers (shortcut for -t UUID)

    -h            Print this help message

genfstab generates output suitable for addition to an fstab file based on the
devices mounted under the mountpoint specified by the given root.

root@archiso ~ # arch-chroot /mnt /bin/bash
[root@archiso /]# _
```

chroot

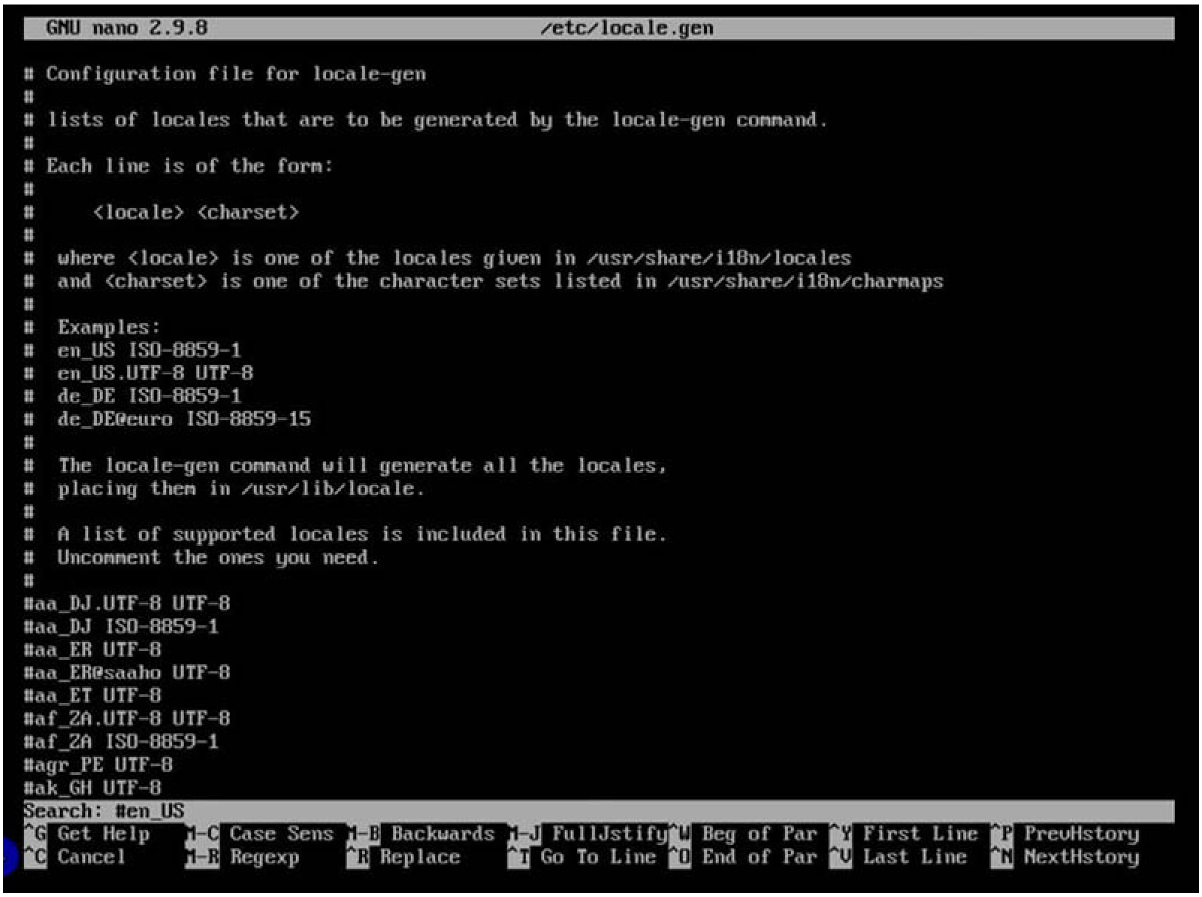
You change your live ISO environment to the root environment of the installed Arch system. This way, you access the system as a root user. A bit later, you will also add the regular user.

Set locale

To set the localization, you will have to work in Nano editor. Type:

and press `Enter`.

Find the language you are going to use. In my case, I am going to install American English. Activate the search option by pressing the shortcut Ctrl + W (the shortcuts are listed at the bottom of the screen) and type `#en_US`. Press `Enter`.



```
 GNU nano 2.9.8                             /etc/locale.gen

# Configuration file for locale-gen
#
# lists of locales that are to be generated by the locale-gen command.
#
# Each line is of the form:
#
#     <locale> <charset>
#
#  where <locale> is one of the locales given in /usr/share/i18n/locales
#  and <charset> is one of the character sets listed in /usr/share/i18n/charmaps
#
#  Examples:
#  en_US ISO-8859-1
#  en_US.UTF-8 UTF-8
#  de_DE ISO-8859-1
#  de_DE@euro ISO-8859-15
#
#  The locale-gen command will generate all the locales,
#  placing them in /usr/lib/locale.
#
#  A list of supported locales is included in this file.
#  Uncomment the ones you need.
#
#aa_DJ.UTF-8 UTF-8
#aa_DJ ISO-8859-1
#aa_ER UTF-8
#aa_ER@saaho UTF-8
#aa_ET UTF-8
#af_ZA.UTF-8 UTF-8
#af_ZA ISO-8859-1
#agr_PE UTF-8
#ak_GH UTF-8
Search: #en_US
^G Get Help   M-C Case Sens M-B Backwards M-J FullJstify^W Beg of Par ^Y First Line ^P PrevHstory
^C Cancel     M-R Regexp    ^R Replace    ^T Go To Line ^O End of Par ^V Last Line  ^N NextHstory
```

Search for #en_US in Nano

You should jump to the line `#en_US.UTF-8 UTF-8`

Uncomment it by removing the `#` sign. Press `Ctrl+O Enter` to save and `Ctrl+X Enter` to exit the editor.

Next, you have to generate the locale. Run:

And create the `locale.conf` with corresponding language settings:

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

Set the time zone

To set the time zone, type:

```
ln -sf /usr/share/zoneinfo/
```

and press the Tab key to see all the available options. In my case, I need to use Europe. Again, you can press the Tab key, and you will see all the available cities. I will use Stockholm. Save this link to `/etc/localtime`. The final command will look as follows:

```
ln -sf /usr/share/zoneinfo/Europe/Stockholm /etc/localtime
```

Instead of Europe and Stockholm, you can select your region and time zone.

Set local time

To set the time for the system, run this command:

And check the time:

If the time is incorrect, go back and make sure you have set the timezone correctly.

Set hostname

A hostname is the computer's name. Let's name it `archPC`. Use the following command:

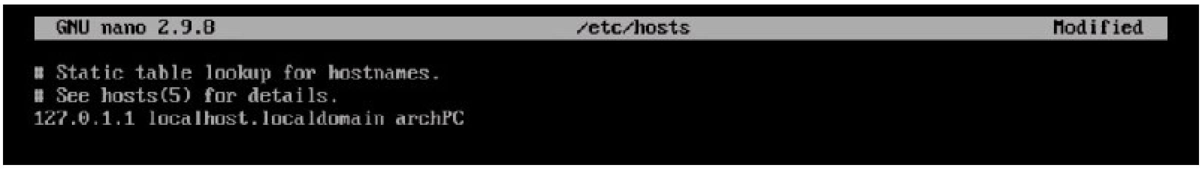
```
echo archPC > /etc/hostname
```

You also need to add this name to the `/etc/hosts` file. Type:

and press `Enter`.

In the Nano editor, add this line at the end of the file:

```
127.0.1.1 localhost.localdomain archPC
```



```
  GNU nano 2.9.8                          /etc/hosts                          Modified

# Static table lookup for hostnames.
# See hosts(5) for details.
127.0.1.1 localhost.localdomain archPC
```

Set hostname

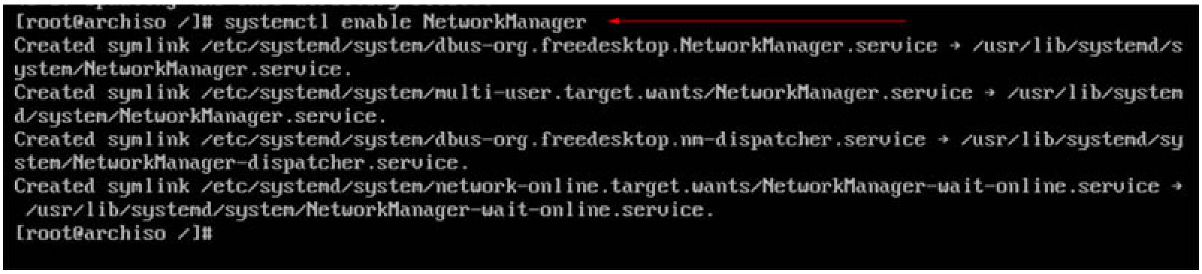
If you use a static IP address, replace `127.0.1.1` with your static IP address given by the Internet provider. Press `Ctrl+O Enter` and `Ctrl+X Enter` to save and exit the editor.

Enable network

First, install the network manager:

Then enable it:

```
systemctl enable NetworkManager
```



```
[root@archiso /]# systemctl enable NetworkManager
Created symlink /etc/systemd/system/dbus-org.freedesktop.NetworkManager.service → /usr/lib/systemd/s
ystem/NetworkManager.service.
Created symlink /etc/systemd/system/multi-user.target.wants/NetworkManager.service → /usr/lib/system
d/system/NetworkManager.service.
Created symlink /etc/systemd/system/dbus-org.freedesktop.nm-dispatcher.service → /usr/lib/systemd/sy
stem/NetworkManager-dispatcher.service.
Created symlink /etc/systemd/system/network-online.target.wants/NetworkManager-wait-online.service →
 /usr/lib/systemd/system/NetworkManager-wait-online.service.
[root@archiso /]#
```

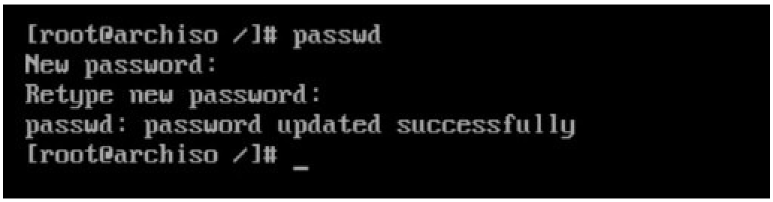
Enable the Network Manager

Now, the system will be able to run a network manager at the system boot and connect to the Internet automatically. Remember, these settings work only for the wired internet connection.

Set root password

Next, set the root password. Type:

and type your password twice. Be attentive, as you will see nothing while typing.



```
[root@archiso /]# passwd
New password:
Retype new password:
passwd: password updated successfully
[root@archiso /]# _
```

Set password

Install GRUB

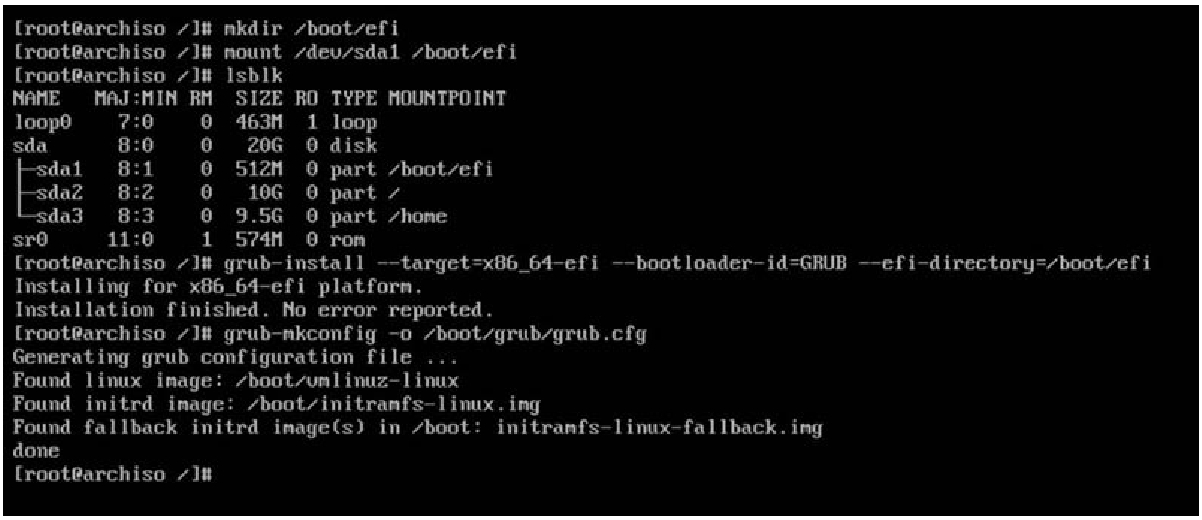
There are several bootloaders you can install on Arch Linux. I recommend the GRUB bootloader. It is the most popular, highly configurable and easy to use bootloader.

Install the GRUB bootloader and EFI boot manager packages:

```
pacman -S grub efibootmgr
```

Next, using these packages, install the bootlader on your system and generate its configuration files by running these commands one by one:

```
mkdir /boot/efi
mount /dev/sda1 /boot/efi
lsblk # to check if everything is mounted correctly
grub-install --target=x86_64-efi --bootloader-id=GRUB --efi-directory=/boot/efi --removable
grub-mkconfig -o /boot/grub/grub.cfg
```



```
[root@archiso /]# mkdir /boot/efi
[root@archiso /]# mount /dev/sda1 /boot/efi
[root@archiso /]# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0  463M  1 loop
sda      8:0    0   20G  0 disk
├─sda1   8:1    0  512M  0 part /boot/efi
├─sda2   8:2    0   10G  0 part /
└─sda3   8:3    0  9.5G  0 part /home
sr0     11:0    1  574M  0 rom
[root@archiso /]# grub-install --target=x86_64-efi --bootloader-id=GRUB --efi-directory=/boot/efi
Installing for x86_64-efi platform.
Installation finished. No error reported.
[root@archiso /]# grub-mkconfig -o /boot/grub/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-linux
Found initrd image: /boot/initramfs-linux.img
Found fallback initrd image(s) in /boot: initramfs-linux-fallback.img
done
[root@archiso /]#
```

GRUB installation

Legacy mode! *If you do the legacy installation, install the GRUB in this way:*

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

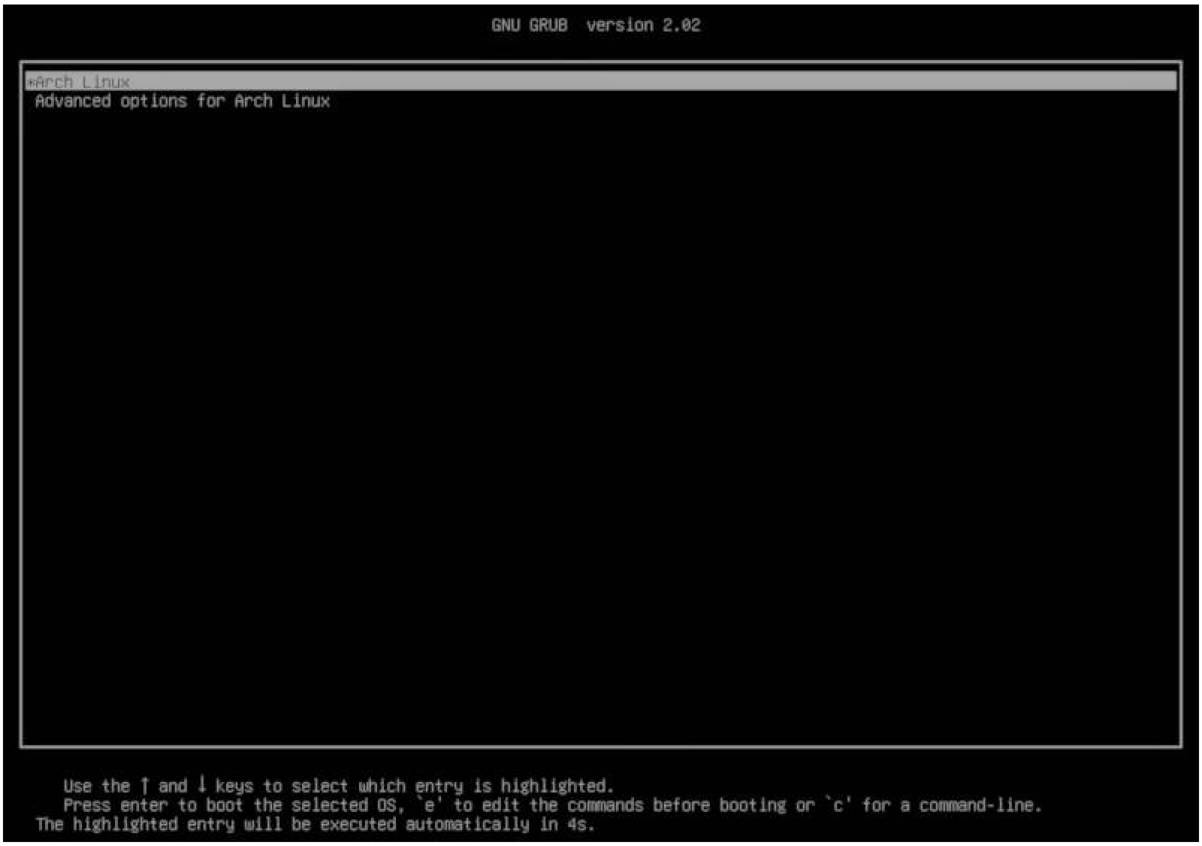
Basically, the minimal installation of Arch Linux is complete.

Reboot

Next, exit from the system root account, unmount all mounted partitions and reboot the system by running these commands:

```
exit
umount -R /mnt
reboot
```

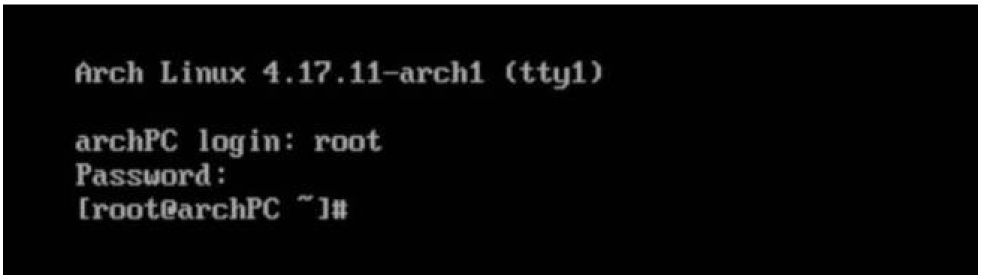
If you did everything correctly, after the reboot, you will see the GRUB welcome screen with Arch Linux installed.



```
                         GNU GRUB  version 2.02

*Arch Linux
 Advanced options for Arch Linux

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, `e' to edit the commands before booting or `c' for a command-line.
 The highlighted entry will be executed automatically in 4s.
```

GRUB screen with Arch Linux

To continue, log in as a root user with a previously set password.



```
Arch Linux 4.17.11-arch1 (tty1)

archPC login: root
Password:
[root@archPC ~]#
```

Log in as a root user

Swap File

You probably have noticed that I have not created a Swap partition. This is because I recommend to use a Swap file. Let's create a swap file for this Arch Linux installation too.

Create a Swap file of `3G` or whatever your RAM size is:

```
fallocate -l 3G /swapfile
```

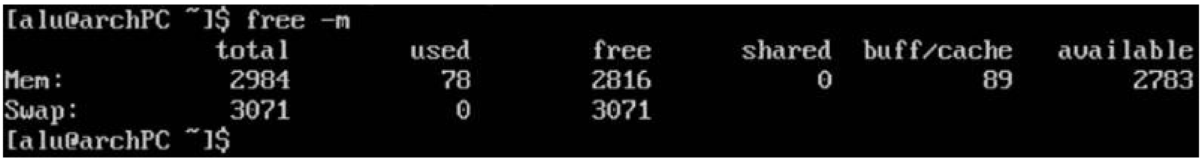
Change its access rules, format and enable it:

```
chmod 600 /swapfile
mkswap /swapfile
swapon /swapfile
```

Also, add this Swap file to the `/etc/fstab`:

```
echo '/swapfile none swap sw 0 0' >> /etc/fstab
```

And check if the Swap file is working:



```
[alu@archPC ~]$ free -m
              total        used        free      shared  buff/cache   available
Mem:           2984          78        2816           0          89        2783
Swap:          3071           0        3071
[alu@archPC ~]$
```

Add user

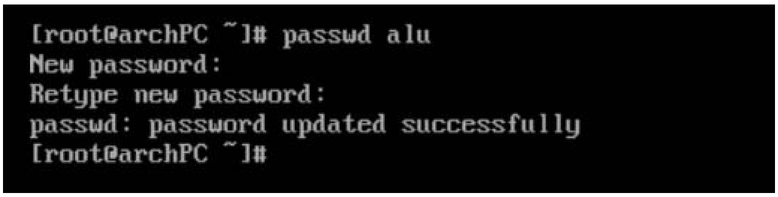
It is not a good idea to constantly work from the root account. So, after the login, create a user account:

```
useradd -m -g users -G wheel -s /bin/bash username
```

Write your name instead of `username`.

Also, create a password for the new user:

Instead of `username`, use the name you created in the previous step. Type the password twice.



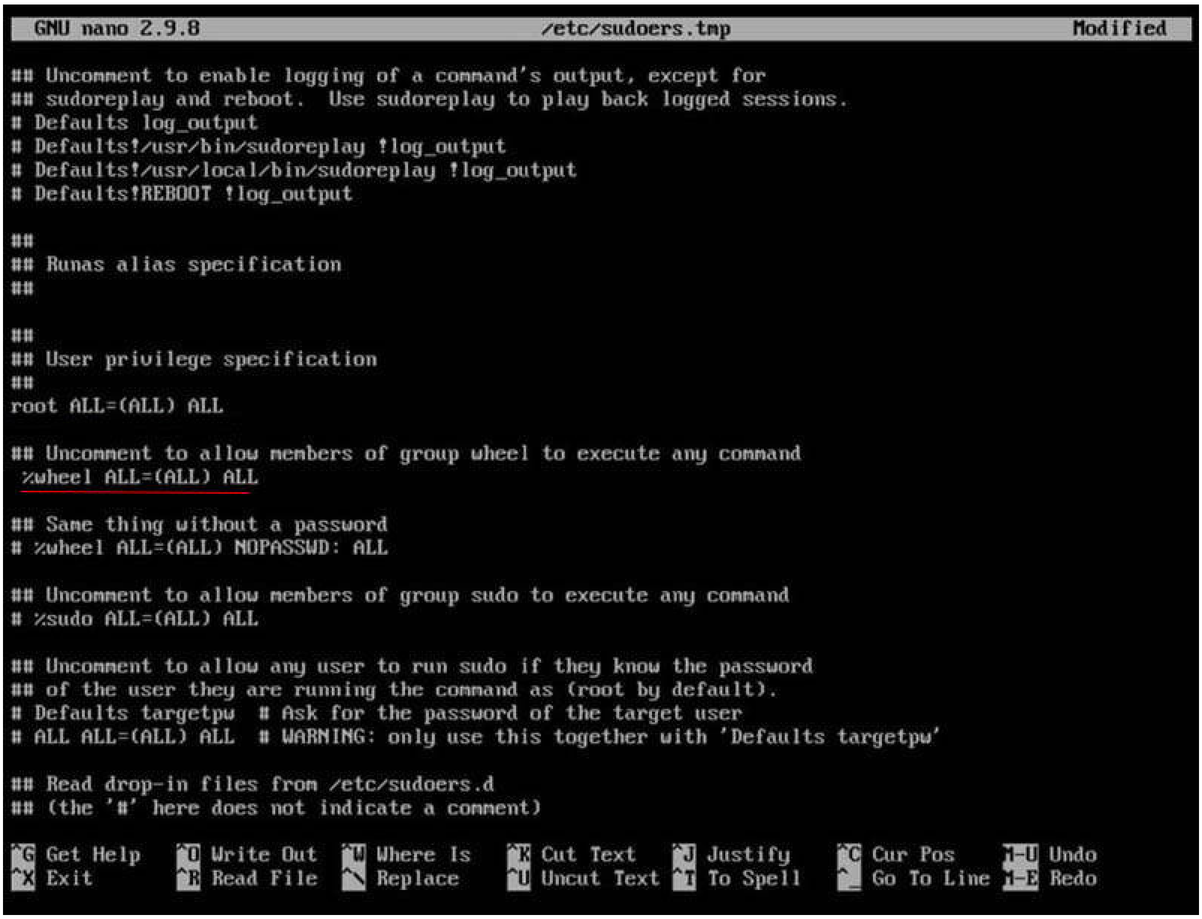
```
[root@archPC ~]# passwd alu
New password:
Retype new password:
passwd: password updated successfully
[root@archPC ~]#
```

Set the user password

Next, enable sudo privileges for a newly created user:

Using the arrow keys, scroll down the screen and find the line:

Uncomment it, by removing the # sign.



```
  GNU nano 2.9.8                    /etc/sudoers.tmp                    Modified

## Uncomment to enable logging of a command's output, except for
## sudoreplay and reboot.  Use sudoreplay to play back logged sessions.
# Defaults log_output
# Defaults!/usr/bin/sudoreplay !log_output
# Defaults!/usr/local/bin/sudoreplay !log_output
# Defaults!REBOOT !log_output

##
## Runas alias specification
##

##
## User privilege specification
##
root ALL=(ALL) ALL

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

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

## Uncomment to allow members of group sudo to execute any command
# %sudo 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  # WARNING: only use this together with 'Defaults targetpw'

## Read drop-in files from /etc/sudoers.d
## (the '#' here does not indicate a comment)
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo
```

Uncomment %wheel ALL=(ALL) ALL

Press `Ctrl+O Enter` to save and `Ctrl+X Enter` to exit the editor.

Now, exit the system by running the command:

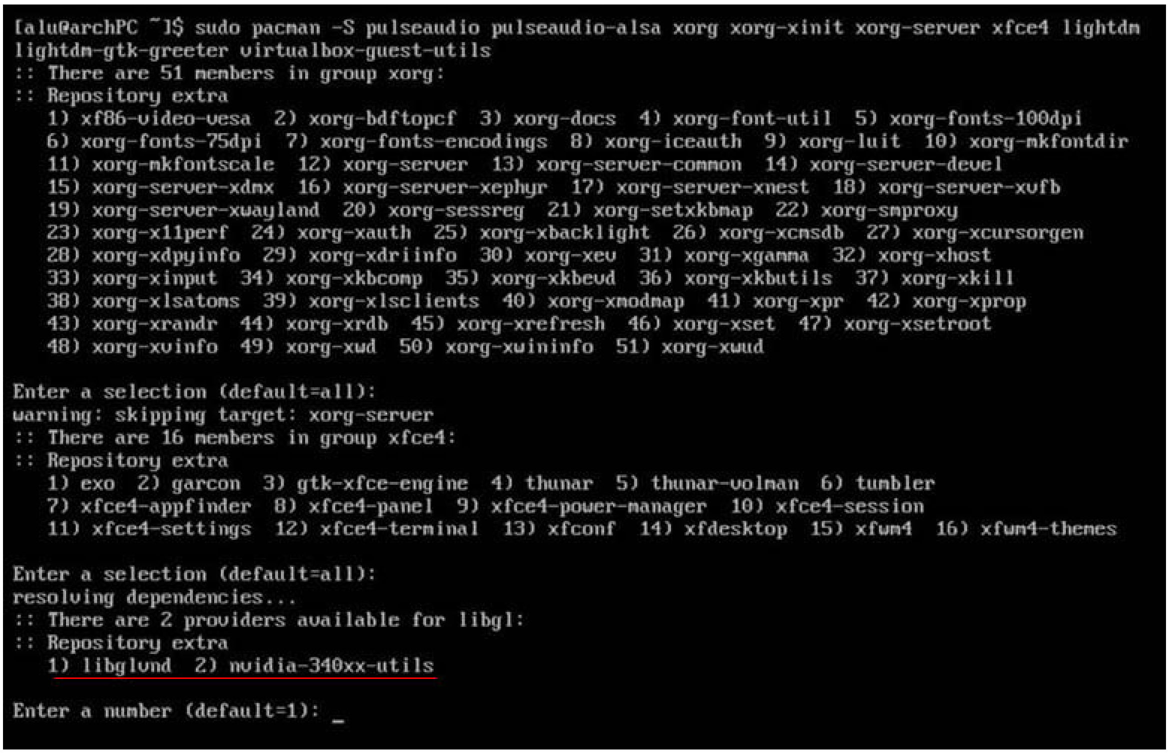
And log in with the regular user credentials which you have just created.

Install X window system and audio

To make the new system usable, install X Window System and audio. I will install XFCE as a desktop environment example.

```
pacman -S pulseaudio pulseaudio-alsa xorg xorg-xinit xorg-server
```

When you press `Enter`, the system will offer to choose the components to install. Just press `Enter` twice to apply the default settings. After that, the system will request to choose the driver for the video card:



```
[alu@archPC ~]$ sudo pacman -S pulseaudio pulseaudio-alsa xorg xorg-xinit xorg-server xfce4 lightdm
lightdm-gtk-greeter virtualbox-guest-utils
:: There are 51 members in group xorg:
:: Repository extra
   1) xf86-video-vesa  2) xorg-bdftopcf  3) xorg-docs  4) xorg-font-util  5) xorg-fonts-100dpi
   6) xorg-fonts-75dpi  7) xorg-fonts-encodings  8) xorg-iceauth  9) xorg-luit  10) xorg-mkfontdir
   11) xorg-mkfontscale  12) xorg-server  13) xorg-server-common  14) xorg-server-devel
   15) xorg-server-xdmx  16) xorg-server-xephyr  17) xorg-server-xnest  18) xorg-server-xvfb
   19) xorg-server-xwayland  20) xorg-sessreg  21) xorg-setxkbmap  22) xorg-smproxy
   23) xorg-x11perf  24) xorg-xauth  25) xorg-xbacklight  26) xorg-xcmsdb  27) xorg-xcursorgen
   28) xorg-xdpyinfo  29) xorg-xdriinfo  30) xorg-xev  31) xorg-xgamma  32) xorg-xhost
   33) xorg-xinput  34) xorg-xkbcomp  35) xorg-xkbevd  36) xorg-xkbutils  37) xorg-xkill
   38) xorg-xlsatoms  39) xorg-xlsclients  40) xorg-xmodmap  41) xorg-xpr  42) xorg-xprop
   43) xorg-xrandr  44) xorg-xrdb  45) xorg-xrefresh  46) xorg-xset  47) xorg-xsetroot
   48) xorg-xvinfo  49) xorg-xwd  50) xorg-xwininfo  51) xorg-xwud

Enter a selection (default=all):
warning: skipping target: xorg-server
:: There are 16 members in group xfce4:
:: Repository extra
   1) exo  2) garcon  3) gtk-xfce-engine  4) thunar  5) thunar-volman  6) tumbler
   7) xfce4-appfinder  8) xfce4-panel  9) xfce4-power-manager  10) xfce4-session
   11) xfce4-settings  12) xfce4-terminal  13) xfconf  14) xfdesktop  15) xfwm4  16) xfwm4-themes

Enter a selection (default=all):
resolving dependencies...
:: There are 2 providers available for libgl:
:: Repository extra
   1) libglvnd  2) nvidia-340xx-utils

Enter a number (default=1): _
```

Arch Linux video driver options

If you have a discrete video graphic card, select the second option. When you use the integrated Intel video card, select the first option. The utility will install many packages. Wait some time until it completes.

If you install the system on VirtualBox, also install guest addition:

```
pacman -S virtualbox-guest-utils
```

Install desktop environment

I install **XFCE** desktop as an example of the graphical interface:

```
pacman -S xfce4 lightdm lightdm-gtk-greeter
echo "exec startxfce4" > ~/.xinitrc
systemctl enable lightdm
```

Xinit file allows to start an Xorg display server automatically.

You can also install other desktops:

Plasma 5:

I showed how to install and configure Plasma 5 in Arch Linux previously.

GNOME:

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

Cinnamon:

```
echo "exec cinnamon-session" > ~/.xinitrc
sudo pacman -S cinnamon mdm
systemctl enable mdm
```

Mate:

```
echo "exec mate-session" > ~/.xinitrc
sudo pacman -S mate lightdm lightdm-gtk-greeter
systemctl enable lightdm
```

Unity:

Unity installation is tricky - see the Arch Linux Wiki.

Budgie:

```
echo "export XDG_CURRENT_DESKTOP=Budgie:GNOME" > ~/.xinitrc
echo "exec budgie-desktop" >> ~/.xinitrc
sudo pacman -S budgie-desktop lightdm lightdm-gtk-greeter
systemctl enable lightdm
```

Openbox:

```
echo "exec openbox-session" > ~/.xinitrc
sudo pacman -S openbox lightdm lightdm-gtk-greeter
systemctl enable lightdm
```

i3:

```
echo "exec i3"  > ~/.xinitrc
sudo pacman -S i3-wm
```

Awesome:

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

Deepin:

```
echo "exec startdde"  > ~/.xinitrc
sudo pacman -S deepin
```

Also, edit the file `/etc/lightdm/lightdm.conf` to have this line:

```
greeter-session=lightdm-deepin-greeter
```

LXDE:

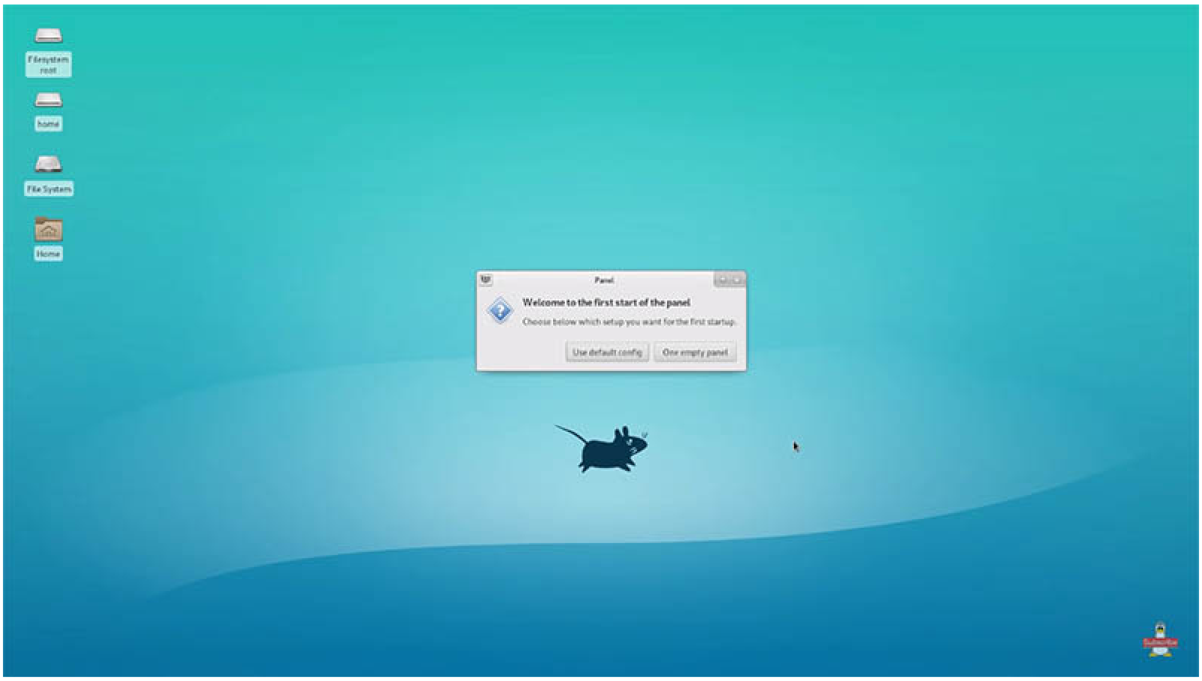
```
echo "exec startlxde"  > ~/.xinitrc
sudo pacman -S lxdm-gtk3 lxdm
```

I would like to point out that I have not tested all these desktops. In particular, some Login Managers may not work with a given desktop. For the full list of Login Managers look at Arch Linux Wiki and try the one you like.

Start GUI

To test whether your graphical environment works, run:

The system must launch the graphical interface.



Panel

Welcome to the first start of the panel

Choose below which setup you want for the first startup.

The default Xfce desktop

To further check that everything works fine, shut down the system and start it again. Hopefully, it will boot again and you will be able to log in.

Arch Linux installation is done!

This is a very minimal installation. From this point, you can install what you want and configure your Arch Linux as you want.

So, this step by step Arch Linux installation guide is over. As you can see, the Arch Linux installation process is a little complicated but manageable. This system is still not complete and you will need to install many more packages and configure it. Nevertheless, the most difficult part is done and you have a good starting point.

Do you want to share your experience or have any questions about how to install Arch Linux? Leave your comment below.