Installation guide

This document is a guide for installing **Arch Linux** using the live system booted from an installation medium made from an official installation image. The installation medium provides accessibility features which are described on the page **Install Arch Linux with accessibility options**. For alternative means of installation, see **Category:Installation process**.

Before installing, it would be advised to view the **FAQ**. For conventions used in this document, see **Help:Reading**. In particular, code examples may contain placeholders (formatted in *italics*) that must be replaced manually.

This guide is kept concise and you are advised to follow the instructions in the presented order per section. For more detailed instructions, see the respective **ArchWiki** articles or the various programs' **man pages**, both linked from this guide. For interactive help, the **IRC channel** and the **forums (https://bbs.archlinux.org/)** are also available.

Arch Linux should run on any **x86_64**-compatible machine with a minimum of 512 MiB RAM, though more memory is needed to boot the live system for installation.**[1] (https://lists.archlin ux.org/archives/list/arch-releng@lists.archlinux.org/message/D5HSGOFTPGYI6IZ UEB3ZNAX4D3F3ID37/)** A basic installation should take less than 2 GiB of disk space. As the installation process needs to retrieve packages from a remote repository, this guide assumes a working internet connection is available.

1 Pre-installation

1.1 Acquire an installation image

Visit the **Download (https://archlinux.org/download/)** page and, depending on how you want to boot, acquire the ISO file or a netboot image, and the respective **GnuPG** signature.

1.2 Verify signature

It is recommended to verify the image signature before use, especially when downloading from an *HTTP mirror*, where downloads are generally prone to be intercepted to **serve malicious images (https://www2.cs.arizona.edu/stork/packagemanagersecurity/attacks-on-pa ckage-managers.html)**.

On a system with **GnuPG** installed, do this by downloading the *ISO PGP signature* (**under Checksums in the page Download (https://archlinux.org/download/#checksums)**) to the ISO directory, and **verifying** it with:

```
$ gpg --keyserver-options auto-key-retrieve --verify archlinux-version-x86_64.iso.sig
```

Alternatively, from an existing Arch Linux installation run:

```
$ pacman-key -v archlinux-version-x86_64.iso.sig
```

Note:

- The signature itself could be manipulated if it is downloaded from a mirror site, instead of from **archlinux.org (https://archlinux.org/download/)** as above. In this case, ensure that the public key, which is used to decode the signature, is signed by another, trustworthy key. The `gpg` command will output the fingerprint of the public key.
- Another method to verify the authenticity of the signature is to ensure that the public key's fingerprint is identical to the key fingerprint of the **Arch Linux developer (https://archlinux. org/people/developers/)** who signed the ISO-file. See **Wikipedia:Public-key cryptography** for more information on the public-key process to authenticate keys.

1.3 Prepare an installation medium

The ISO can be supplied to the target machine via a **USB flash drive**, an **optical disc** or a network with **PXE**: follow the appropriate article to prepare yourself an installation medium from the ISO file.

For the netboot image, follow **Netboot#Boot from a USB flash drive** to prepare a USB flash drive for UEFI booting.

1.4 Boot the live environment

Note: Arch Linux installation images do not support Secure Boot. You will need to **disable Secure Boot** to boot the installation medium. If desired, **Secure Boot** can be set up after completing the installation.

1. Point the current boot device to the one which has the Arch Linux installation medium. Typically it is achieved by pressing a key during the **POST** phase, as indicated on the splash screen. Refer to your motherboard's manual for details.
2. When the installation medium's boot loader menu appears,

   - if you used the ISO, select *Arch Linux install medium* and press `Enter` to enter the installation environment.
   - if you used the Netboot image, choose a geographically close mirror from *Mirror* menu, then select *Boot Arch Linux* and press `Enter`.

     > **Tip:**
     >
     > - The ISO uses **GRUB** for UEFI and **syslinux** for BIOS booting. Use respectively `e` or `Tab` to enter the **boot parameters**. The Netboot image uses iPXE and the boot parameters can be specified in the *Boot options* menu. See **README.bootparams (https://gitlab.archlinux.org/archlinux/mkinitcpio/mkinitcpio-archiso/blob/mast er/docs/README.bootparams)** for a list.
     > - A common example of manually defined boot parameter would be the font size. For better readability on HiDPI screens—when they are not already recognized as such —using `fbcon=font:TER16x32` can help. See **HiDPI#Linux console (tty)** for a detailed explanation.

3.
4. You will be logged in on the first **virtual console** as the root user, and presented with a **Zsh** shell prompt.

To switch to a different console—for example, to view this guide with **Lynx (https://lynx.invisi ble-island.net/lynx_help/Lynx_users_guide.html)** alongside the installation—use the `Alt+`*arrow* **shortcut**. To **edit** configuration files, **mcedit(1) (https://man.archlinux.org/**

man/mcedit.1), **nano** and **vim** are available. See **pkglist.x86_64.txt (https://geo.mirror.p kgbuild.com/iso/latest/arch/pkglist.x86_64.txt)** for a list of the packages included in the installation medium.

1.5 Set the console keyboard layout and font

The default **console keymap** is **US**. Available layouts can be listed with:

```
# localectl list-keymaps
```

To set the keyboard layout, pass its name to **loadkeys(1) (https://man.archlinux.org/man/ loadkeys.1)**. For example, to set a **German** keyboard layout:

```
# loadkeys de-latin1
```

Console fonts are located in `/usr/share/kbd/consolefonts/` and can likewise be set with **setfont(8) (https://man.archlinux.org/man/setfont.8)** omitting the path and file extension. For example, to use one of the largest fonts suitable for **HiDPI screens**, run:

```
# setfont ter-132b
```

1.6 Verify the boot mode

To verify the boot mode, check the UEFI bitness:

```
# cat /sys/firmware/efi/fw_platform_size
```

If the command returns `64`, then system is booted in UEFI mode and has a 64-bit x64 UEFI. If the command returns `32`, then system is booted in UEFI mode and has a 32-bit IA32 UEFI; while this is supported, it will limit the boot loader choice to systemd-boot. If the file does not exist, the system may be booted in **BIOS** (or **CSM**) mode. If the system did not boot in the mode you desired (UEFI vs BIOS), refer to your motherboard's manual.

1.7 Connect to the internet

To set up a network connection in the live environment, go through the following steps:

- Ensure your **network interface** is listed and enabled, for example with **ip-link(8) (http s://man.archlinux.org/man/ip-link.8)**:

  ```
  # ip link
  ```

- For wireless and WWAN, make sure the card is not blocked with **rfkill**.
- Connect to the network:
  - Ethernet—plug in the cable.
  - Wi-Fi—authenticate to the wireless network using **iwctl**.
  - Mobile broadband modem—connect to the mobile network with the **mmcli** utility.
- Configure your network connection:

- **DHCP**: dynamic IP address and DNS server assignment (provided by **systemd-networkd** and **systemd-resolved**) should work out of the box for Ethernet, WLAN, and WWAN network interfaces.
  - Static IP address: follow **Network configuration#Static IP address**.
- The connection may be verified with **ping**:

```
# ping archlinux.org
```

Note: In the installation image, **systemd-networkd**, **systemd-resolved**, **iwd** and **ModemManager** are preconfigured and enabled by default. That will not be the case for the installed system.

1.8 Update the system clock

In the live environment **systemd-timesyncd** is enabled by default and time will be synced automatically once a connection to the internet is established.

Use **timedatectl(1) (https://man.archlinux.org/man/timedatectl.1)** to ensure the system clock is accurate:

```
# timedatectl
```

1.9 Partition the disks

When recognized by the live system, disks are assigned to a **block device** such as /dev/sda , /dev/nvme0n1 or /dev/mmcblk0 . To identify these devices, use **lsblk** or *fdisk*.

```
# fdisk -l
```

Results ending in rom , loop or airootfs may be ignored. mmcblk* devices ending in rpbm , boot0 and boot1 can be ignored.

Note: If the disk does not show up, **make sure the disk controller is not in RAID mode**.

Tip: Check that your NVMe drives and Advanced Format hard disk drives are using the **optimal logical sector size** before partitioning.

The following **partitions** are **required** for a chosen device:

- One partition for the **root directory** / .
- For booting in **UEFI** mode: an **EFI system partition**.

Use a **partitioning tool** like **fdisk** to modify partition tables. For example:

```
# fdisk /dev/the_disk_to_be_partitioned
```

Note:

- Take time to plan a long-term partitioning scheme to avoid risky and complicated conversion or re-partitioning procedures in the future.
- If you want to create any stacked block devices for **LVM**, **system encryption** or **RAID**, do it now.
- If the disk from which you want to boot **already has an EFI system partition**, do not create another one, but use the existing partition instead.
- **Swap** space can be set on a **swap file** for file systems supporting it.

1.9.1 Example layouts

UEFI with **GPT**

Mount point	Partition	Partition type	Suggested size
`/mnt/boot` [1]	`/dev/`*`efi_system_partition`*	**EFI system partition**	1 GiB
`[SWAP]`	`/dev/`*`swap_partition`*	Linux swap	At least 4 GiB
`/mnt`	`/dev/`*`root_partition`*	Linux x86-64 root (/)	Remainder of the device. At least 23–32 GiB.

1. **Other mount points**, such as `/mnt/efi` , are possible, provided that the used boot loader is capable of loading the kernel and initramfs images from the root volume. See the warning in **Arch boot process#Boot loader**.

BIOS with **MBR**

Mount point	Partition	Partition type	Suggested size
`[SWAP]`	`/dev/`*`swap_partition`*	Linux swap	At least 4 GiB
`/mnt`	`/dev/`*`root_partition`*	Linux	Remainder of the device. At least 23–32 GiB.

See also **Partitioning#Example layouts**.

1.10 Format the partitions

Once the partitions have been created, each newly created partition must be formatted with an appropriate **file system**. See **File systems#Create a file system** for details.

For example, to create an **Ext4** file system on `/dev/`*`root_partition`* , run:

```
# mkfs.ext4 /dev/root_partition
```

If you created a partition for **swap**, initialize it with **mkswap(8) (https://man.archlinux.org/man/mkswap.8)**:

```
# mkswap /dev/swap_partition
```

Note: For stacked block devices replace `/dev/*_partition` with the appropriate block device path.

If you created an EFI system partition, **format it** to FAT32 using **mkfs.fat(8) (https://man.archlinux.org/man/mkfs.fat.8)**.

Warning: Only format the EFI system partition if you created it during the partitioning step. If there already was an EFI system partition on disk beforehand, reformatting it can destroy the

boot loaders of other installed operating systems.

```
# mkfs.fat -F 32 /dev/efi_system_partition
```

1.11 Mount the file systems

Mount the root volume to `/mnt` . For example, if the root volume is `/dev/root_partition` :

```
# mount /dev/root_partition /mnt
```

Create any remaining mount points (such as `/mnt/boot`) and mount the volumes in their corresponding hierarchical order.

> **Tip:** Run **mount(8) (https://man.archlinux.org/man/mount.8)** with the `--mkdir`
> option to create the specified mount point. Alternatively, create it using **mkdir(1) (https://m
> an.archlinux.org/man/mkdir.1)** beforehand.

For UEFI systems, mount the EFI system partition:

```
# mount --mkdir /dev/efi_system_partition /mnt/boot
```

If you created a **swap** volume, enable it with **swapon(8) (https://man.archlinux.org/man/sw
apon.8)**:

```
# swapon /dev/swap_partition
```

genfstab(8) (https://man.archlinux.org/man/genfstab.8) will later detect mounted file systems and swap space.

2 Installation

2.1 Select the mirrors

Packages to be installed must be downloaded from **mirror servers**, which are defined in `/etc/pacman.d/mirrorlist` . On the live system, after connecting to the internet, **reflector** updates the mirror list by choosing 20 most recently synchronized HTTPS mirrors and sorting them by download rate.

The higher a mirror is placed in the list, the more priority it is given when downloading a package. You may want to inspect the file to see if it is satisfactory. If it is not, **edit** the file accordingly, and move the geographically closest mirrors to the top of the list, although other criteria should be taken into account.

This file will later be copied to the new system by *pacstrap*, so it is worth getting right.

2.2 Install essential packages

> **Note:** No software or configuration (except for `/etc/pacman.d/mirrorlist`) gets carried over from the live environment to the installed system.

Use the **pacstrap(8) (https://man.archlinux.org/man/pacstrap.8)** script to install the **base (https://archlinux.org/packages/?name=base)** package, Linux **kernel** and firmware for common hardware:

```
# pacstrap -K /mnt base linux linux-firmware
```

> **Tip:**
> - You can substitute **linux (https://archlinux.org/packages/?name=linux)** with a **kernel** package of your choice, or you could omit it entirely when installing in a **container**.
> - You could omit the installation of the firmware package when installing in a virtual machine or container.

The **base (https://archlinux.org/packages/?name=base)** package does not include all tools from the live installation, so installing more packages may be necessary for a fully functional base system. To install other packages or package groups, append the names to the *pacstrap* command above (space separated) or use **pacman** to **install** them while **chrooted into the new system**. In particular, consider installing:

- CPU **microcode** updates—**amd-ucode (https://archlinux.org/packages/?name=amd-ucode)** or **intel-ucode (https://archlinux.org/packages/?name=intel-ucode)**—for hardware bug and security fixes,
- **userspace utilities for file systems** that will be used on the system—for the purposes of e.g. file system creation and **fsck**,
- utilities for accessing and managing **RAID** or **LVM** if they will be used on the system,
- specific firmware for other devices not included in **linux-firmware (https://archlinux.org/packages/?name=linux-firmware)** (e.g. **sof-firmware (https://archlinux.org/packages/?name=sof-firmware)** for **onboard audio**, **linux-firmware-marvell (https://archlinux.org/packages/?name=linux-firmware-marvell)** for Marvell wireless and any of the multiple firmware packages for **Broadcom wireless**),
- software necessary for **networking** (e.g. **a network manager or a standalone DHCP client**, **authentication software** for Wi-Fi, **ModemManager** for mobile broadband connections),
- a **text editor**,
- packages for accessing documentation in **man** and **info** pages: **man-db (https://archlinux.org/packages/?name=man-db)**, **man-pages (https://archlinux.org/packages/?name=man-pages)** and **texinfo (https://archlinux.org/packages/?name=texinfo)**.

For comparison, packages available in the live system can be found in **pkglist.x86_64.txt (http s://geo.mirror.pkgbuild.com/iso/latest/arch/pkglist.x86_64.txt)**.

3 Configure the system

3.1 Fstab

Generate an **fstab** file (use `-U` or `-L` to define by **UUID** or labels, respectively):

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

Check the resulting `/mnt/etc/fstab` file, and **edit** it in case of errors.

3.2 Chroot

Change root into the new system:

```
# arch-chroot /mnt
```

3.3 Time

Set the **time zone**:

```
# ln -sf /usr/share/zoneinfo/Region/City /etc/localtime
```

Run **hwclock(8) (https://man.archlinux.org/man/hwclock.8)** to generate `/etc/adjtime`:

```
# hwclock --systohc
```

This command assumes the hardware clock is set to **UTC**. See **System time#Time standard** for details.

To prevent clock drift and ensure accurate time, set up **time synchronization** using a **Network Time Protocol** (NTP) client such as **systemd-timesyncd**.

3.4 Localization

Edit `/etc/locale.gen` and uncomment `en_US.UTF-8 UTF-8` and other needed UTF-8 **locales**. Generate the locales by running:

```
# locale-gen
```

Create the **locale.conf(5) (https://man.archlinux.org/man/locale.conf.5)** file, and **set the LANG variable** accordingly:

```
/etc/locale.conf
```

```
LANG=en_US.UTF-8
```

If you **set the console keyboard layout**, make the changes persistent in **vconsole.conf(5) (https://man.archlinux.org/man/vconsole.conf.5)**:

```
/etc/vconsole.conf
```
```
KEYMAP=de-latin1
```

3.5 Network configuration

Create the **hostname** file:

```
/etc/hostname
```
```
yourhostname
```

Complete the **network configuration** for the newly installed environment. That may include installing suitable **network management** software, configuring it if necessary and enabling its systemd unit so that it starts at boot.

3.6 Initramfs

Creating a new *initramfs* is usually not required, because **mkinitcpio** was run on installation of the **kernel** package with *pacstrap*.

For **LVM**, **system encryption** or **RAID**, modify **mkinitcpio.conf(5) (https://man.archlinux.org/man/mkinitcpio.conf.5)** and recreate the initramfs image:

```
# mkinitcpio -P
```

3.7 Root password

Set the root **password**:

```
# passwd
```

3.8 Boot loader

Choose and install a Linux-capable **boot loader**.

4 Reboot

Exit the chroot environment by typing `exit` or pressing `Ctrl+d`.

Optionally manually unmount all the partitions with `umount -R /mnt`: this allows noticing any "busy" partitions, and finding the cause with **fuser(1) (https://man.archlinux.org/man/fuser.1)**.

Finally, restart the machine by typing `reboot` : any partitions still mounted will be automatically unmounted by *systemd*. Remember to remove the installation medium and then login into the new system with the root account.

5 Post-installation

See **General recommendations** for system management directions and post-installation tutorials (like creating unprivileged user accounts, setting up a graphical user interface, sound or a touchpad).

For a list of applications that may be of interest, see **List of applications**.

Retrieved from "**https://wiki.archlinux.org/index.php?title=Installation_guide&oldid=805279**"

- ■