

Arch linux BIOS 安装教程

本教程使用的Arch Linux镜像是10.01的镜像，下载后请将.jpg的后缀名删除掉，不会的请自行Bing

[Arch Linux 2022.10.01镜像下载地址](#)

本教程大量借鉴[Microsoft Xiaoice的legacy BIOS教程](#)

Arch linux是一个滚动更新的Linux发行版，它具有一下的特点

简洁

Arch Linux 将简洁定义为：**避免任何不必要的添加、修改和复杂增加**。它提供的软件都来自原始开发者（[上游](#)），仅进行和发行版（下游）相关的最小修改。

- 不包含上游不愿意接受的补丁。绝大部分 Arch 下游补丁已经被上游接受，下一个正式版本里会包含。
- 配置文件也是来自上游，仅包含发行版必须的调整，比如特殊的文件系统路径变动。Arch 不会在安装一个软件包后就自动启动服务。
- 软件包通常都和一个上游项目直接对应。仅在极少数情况下才会拆分软件包。
- 官方不支持图形化配置界面，建议用户使用命令行或文本编辑器修改设置。

现代

1.1 Arch尽全力保持软件处于最新的稳定版本，只要不出现系统软件包破损，都尽量用最新版本。Arch采用[滚动升级](#)策略，安装之后可以持续升级。

Arch向[GNU/Linux](#)用户提供了许多新特性，包括[systemd](#)初始化系统、现代的[文件系统](#)、LVM2/EVMS、软件磁盘阵列（软RAID）、udev支持、initcpio（附带mkinitcpio）以及最新的内核。

实用

1.2 Arch 注重实用性，避免意识形态之争。最终的设计决策都是由开发者的共识决定。开发者依赖于事实的技术分析和讨论，避免政治因素，不会被流行观点左右。Arch Linux 的仓库中包含大量的软件包和编译脚本。用户可以按照需要进行自由选择。仓库中既提供了开源、自由的软件，也提供了闭源软件。**实用性大于意识形态**。

以用户为中心

许多 Linux 发行版都试图变得更“用户友好”，Arch Linux 则一直是，永远会是“以用户为中心”。此发行版是为了满足贡献者的需求，而不是为了吸引尽可能多的用户。Arch 适用于乐于自己动手的用户，他们愿意花时间阅读文档，解决自己的问题。

Arch 鼓励每一个用户 [参与](#) 和贡献，报告和帮助修复 [bugs](#)，提供软件包补丁和参加核心 [项目](#)：Arch 开发者都是志愿者，通过持续的贡献成为团队的一员。Archers 可以自行贡献软件包到 [Arch 用户仓库](#)，提升 [ArchWiki 文档质量](#)，在 [论坛](#)、[邮件列表](#) 或者 [IRC](#) 中给其它用户提供技术支持。Arch Linux 是全球很多用户的选择，已经有很多[国际社区](#)提供帮助和文档翻译。

——[Arch Wiki](#)

0.终端编辑器 vim 的使用

在安装 Arch Linux 的过程中，需要使用到 vim 编辑器，如果你不会使用，这里首先进行一个简要的介绍，你需要掌握如下基本操作。实践的环境可以找一个在线的 Linux 体验环境进行 vim 的尝试，如copy.sh。注意，由于其是在线环境，所以性能较差，执行 vim 命令时需要耐心等待。

```
vim 1.txt    #创建并编辑名为1.txt的文件
```

你可以看到进入了一个空的界面。此时你处在 vim 的 **命令模式**。在 **命令模式** 下，可以用一些 > 快捷指令来对文本进行操作。现在我们输入 **a** 进入 vim 的 **编辑模式**，此时输入任意文本，即 > 可进行编辑。在输入完成后，我们按下 Esc 键，即可从 **编辑模式** 退出到 **命令模式**。此时输入 **:wq** 即可保存并退出 vim。下面介绍一些在命令模式下常用的命令

```
:wq      # 保存退出
:q!      # 不保存，强制退出
dd       # 删除一行
2dd      # 删除两行
gg       # 回到文本第一行
shift+g  # 转到文本最后一行
/xxx     # 在文中搜索内容 'xxx' 回车搜索，按n键转到下一个
?xxx     # 反向搜索
```

拓展链接：需要完整教程的读者可以在终端中输入命令 **vimtutor** 来学习完整的 vim 教程。

1. 确保网络环境

如果你可以使用路由器分接出来的网线，以 dhcp 的方式直接上网，那么不用准备什么。如果你的环境只能使用无线网络安装，需要事先把自己所用的 wifi 名称改成自己能记住的英文名称。因为 **安装时无法显示和输入中文名的 wifi**，你会看到一堆不知道是什么的方块，并且在安装过程中你将 > 没有办法输入中文的无线名称进行连接。虽然通过一些繁琐的步骤可以解决终端中文的问题，但是 > 显然这么做在安装 Arch Linux 时毫无必要。

其次，有些笔记本电脑上存在无线网卡的硬件开关或者键盘控制，开机后安装前需要 **确保你的无线网卡硬件开关处于打开状态**。

2. 刻录启动优盘

准备一个 2G 以上的优盘，刻录一个安装启动盘。安装镜像 iso 在 [下载页面](#) 下载，注意，你需要选择最新的镜像下载，选择通过磁力链接或 > torrent 下载，下载完成后，还需要在 archlinux 下载页面下载 **PGP signature** 签名文件(不要从 > 镜像源下载签名文件)，将签名文件和 iso 镜像置于同一文件夹，随后进行对镜像的签名校验，以保 > 证下载的镜像是完整，无错误的，未被篡改的。若你使用 Linux，执行以下命令，确保输出完好的签 > 名。具体镜像名根据名字自行修改。如果你使用其他系统，请自行搜索验证签名的方式。

```
gpg --keyserver-options auto-key-retrieve --verify archlinux-202x.0x.01-x86_64.iso.sig
```

注意，这里的签名校验 **非常重要**，这可以保证你的安装镜像是未被篡改的，同时可以保证你在使用安装盘安装系统时，用正确的公钥校验安装包。

Windows 下推荐使用 [ventoy](#) 或者 [Rufus](#) 或者 [etcher](#) 进行优盘刻录。三者皆为自由软件。具体操作请自行查阅，都非常简单。

Linux 下可以直接用 dd 命令进行刻录。注意 of 的参数为 sdx，不是 sdx1 sdx2 等。

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

bs=4M 指定一个较为合理的文件输入输出块大小。

status=progress 用来输出刻录过程总的信息。

oflag=sync 用来控制写入数据时的行为特征。确保命令结束时数据及元数据真正写入磁盘，而不是刚写入缓存就返回。

3.进入主板 BIOS 进行设置

插入优盘并开机。在开机的时候，按下 F2/F8/F10/DEL 等(取决与你的主板型号，具体请查阅你主板的相关信息)按键，进入主板的 BIOS 设置界面。

4.关闭主板设置中的 Secure Boot

在类似名为 `security` 的选项卡中，找到一项名为 Secure Boot(名称可能略有差异)的选项，选择 Disable 将其禁用。

5.调整硬盘启动顺序

在类似名为 `boot` 的选项卡中，找到类似名为 Boot Options(名称可能略有差异)的设置选项，将 USB 优盘的启动顺序调至首位。

——[Arch linux studio \(需科学\)](#)

#

教程开始：

1.下载镜像准备

[点击此处下载系统镜像](#)并[校验](#)镜像的完整性。校验无误后，使用[Ventoy](#)制作安装介质。（如果是首次使用Ventoy，[请点击此处](#)查看使用方法）然后进入BIOS，将安全启动和快速启动关闭并将安装介质（如U盘）设置为第一启动项，并按F10保存退出。进入ArchLinux安装盘即可。

如有疑问请参考Wiki以下章节：

[获取安装映像](#)

[准备安装映像](#)

[启动到Live环境](#)[\]\(Installation guide \(简体中文\) - ArchWiki\)](#)

2.进入Arch linux live CD后连接网络

小技巧：在输入命令时可以与tab键结合，可以是命令出现更少的错误

```
$ systemctl stop reflector.service
#关闭自动设置镜像源

$ nano /etc/pacman.d/mirrorlist
#自己设置镜像源,一般选择中科大(ustc)或清华(tuna)

$ timedatectl set-ntp true
#同步时间

$ timedatectl status
#检查服务状态
```

如有疑问请参考Wiki以下章节:

[验证引导模式](#)

[连接到因特网](#)

[选择镜像和reflector](#)

[更新系统时间](#)

2.1 nano的用法 (高手也可以使用vim)

nano是一个文本编辑器, 对新手来说比较友好。

Q1) 编辑好的文本如何保存?

A1) 按住CTRL键并按一下O键以后在屏幕下方的就是保存文件的位置, 保持默认, 回车。

Q2) 保存好文本以后如何退出nano编辑器?

A2) 按住CTRL键并按一下X键就可以退出了

3.分区&挂载

以下的分区和挂载的相关步骤是以SATA协议的硬盘为例 (其他协议的硬盘请详见Arch Wiki)

```
$ lsblk
#查看硬盘名称 (一般为sda; 其中rom,loop或airoot可以忽略; 这里以sda为例

$fdisk /dev/sdx
#这里的x是指sd后面的一个英文字母
#步骤如下:
Command (m for help): o #输入o新建MBR分区表
Command (m for help): n #输入n创建新分区
Select (default p): p #这里按Enter键创建主分区 (如果想创建逻辑扩展分区请输入e)
Partition number (1-4, default 1): #这里按Enter键
First sector (2048-X, default 2048): #这里按Enter键
Last sector, +/-sectors or +/-size[K,M,G,T,P] (2048-X, default X): +10G #输入+10G
Command (m for help): t #输入t更改分区类型
Hex code or alias (type L to list all): 82 #输入82, 创建swap分区
Command (m for help): n #输入n创建新分区, 然后一直按Enter键, 把剩下的空间全部分配给/分区
Command (m for help): w #输入w写入
```

注：这里的y指的是数字

```
$ lsblk
```

#查看分区结构是否正确

```
$ mkfs.btrfs /dev/sdxy
```

#将sdxy格式化为btrfs

```
$ mkswap /dev/sdaxy
```

#将sdxy设置为swap（跟电脑的内存一样的用途，但速度慢很多）

3.1挂载

```
$ mount -t btrfs -o compress=zstd /dev/sdxy /mnt
```

#将格式为btrfs挂载到/mnt目录并创建btrfs子卷

#步骤如下：

```
btrfs subvolume create /mnt/@ #创建/目录的btrfs子卷
```

```
btrfs subvolume create /mnt/@root #创建/root目录的btrfs子卷
```

```
btrfs subvolume create /mnt/@home #创建/home目录的btrfs子卷
```

```
umount /mnt #卸载/mnt目录
```

```
mount -t btrfs -o subvol=@,compress=zstd /dev/sdxy /mnt #挂载/目录的btrfs子卷到/mnt目录
```

```
mkdir /mnt/root #创建/mnt/root目录
```

```
mount -t btrfs -o subvol=@root,compress=zstd /dev/sda2 /mnt/root #挂载/root目录的btrfs子卷到/mnt/root
```

目录

```
mkdir /mnt/home #创建/mnt/home目录
```

```
mount -t btrfs -o subvol=@home,compress=zstd /dev/sda2 /mnt/home #挂载/home目录的btrfs子卷到/mnt/home目录
```

```
$ swapon /dev/sda1
```

#激活sda1为交换分区

```
$ lsblk -f
```

#查看相应分区的文件系统及挂载是否正确

4.使用pacstrap来安装系统

```
$ vim /etc/pacman.d/mirrorlist
```

#使用vim来编辑镜像源

```
$nano /etc/pacman.d/mirrorlist
```

#使用nano来编辑镜像源

5.安装基础的软件包和初步配置

```
$ pacstrap /mnt linux linux-firmware linux-headers base base-devel vim nano bash-completion networkmanager btrfs-progs pacman-contrib
#安装Linux等软件包，nano可以不选择下载（不会用vim的可以不安装）

$genfstab -U /mnt > /mnt/etc/fstab
#生成/mnt/etc/fstab文件（注意大小写）

$cat /mnt/etc/fstab
#查看/mnt/etc/fstab文件是否正确（如果不正确，请重新分区、挂载、pacstrap）
```

5.1 进入目标系统

```
$ arch-chroot /mnt
#进入目标系统

$ pacman -S grub intel-ucode amd-ucode
#安装grub引导程序&AMD或英特尔微码（AMD CPU安装amd-ucode 英特尔 CPU安装intel-ucode）

$grub-install --target=i386-pc --recheck /dev/sdx
#将grub写入sdx 注：这里的sdx不用加数字！！

$ grub-mkconfig -o /boot/grub/grub.cfg
#生成grub.cfg文件 注：这一步不能忽略，否则将无法启动！！
```

5.2启动网络管理器服务（联网）

```
$ systemctl enable NetworkManager
#设置开机自启NetworkManager服务（注意大小写）

$ passwd root
#设置root密码（输入该命令后需要输入你的密码，但输入的密码是不会显示的，输入完成后回车，再次输入一次与上一个密码，同样是不显示的，回车后即可完成密码设置）
```

如有疑问请参考Wiki以下章节：

[安装必需的软件包](#)

[Fstab](#)

[Chroot](#)

[安装引导程序](#)

[Root密码](#)

6.退出目标系统

```
$ exit
#退出目标系统

$ umount -R /mnt
#卸载/mnt目录

$ reboot
#重启并登录root 注：重新启动前先拔掉U盘
```

7.系统配置

```
$ nmtui
#运行nmtui（根据图形界面提示进行联网操作即可；台式机可跳过）

$ ping archlinux.org
#检查网络连接（如果不停的有输出内容，即为联网成功；按Ctrl+C终止输出）

$ vim /etc/hostname
#创建/etc/hostname文件，加入以下内容：
Arch Linux（可以更改为你喜欢的主机名）
#将主机名设置为Arch Linux

$ vim /etc/hosts
#编辑/etc/hosts文件，在末尾加入以下内容：
127.0.0.1 localhost
::1 localhost
127.0.1.1 arch.localdomain arch
#配置hosts文件，映射IP地址和主机名

$ timedatectl set-timezone Asia/Shanghai && ln -sf
/usr/share/zoneinfo/Asia/Shanghai /etc/localtime && hwclock --systohc
#设置时区为上海（注意大小写）

$ timedatectl set-ntp true
#同步时间

$ timedatectl status
#检查服务状态

$ vim /etc/environment
#编辑/etc/environment文件，在开头加入以下内容：
export EDITOR=/usr/bin/vim
#设置vim为默认文本编辑器（注意大小写）

$ reboot
#重启并登陆root

$ useradd -m -G wheel arch
#添加普通用户，用户名为arch并将arch添加到wheel组中

$ passwd arch
#设置arch密码（注意事项请参考第三阶段的设置root密码部分）

$ id arch
```

#查看用户组是否添加到相应的组中

\$ sudo

#设置用户权限, 删除%wheel ALL=(ALL:ALL) ALL前面的#

\$ reboot

#重启并登陆root

\$ vim /etc/locale.gen

#编辑/etc/locale.gen文件, 删除en_US.UTF-8 UTF-8和zh_CN.UTF-8 UTF-8前面的#

\$ locale-gen

#生成语言

\$ vim /etc/locale.conf

#编辑/etc/locale.conf文件, 在末尾加入以下内容:

LANG="en_US.UTF-8"

#设置语言为en_US.UTF-8, 不要设置为zh_CN.UTF-8 (注意大小写)

\$ reboot

#重启并登陆root

\$ vim /etc/pacman.conf

#编辑/etc/pacman.conf文件, 删除[multilib]区域的所有# (开启32位支持) 并在末尾加入以下内容:
[archlinuxcn]

Server = https://mirrors.ustc.edu.cn/archlinuxcn/\$arch

#添加archlinuxcn源 (一般推荐使用中科大源; 除了可以添加archlinuxcn源外, 还可以添加arch4edu源、blackarch源以及各种私人源, 后面会提到; 注意大小写)

\$ pacman -Sy

#同步数据

\$ pacman -S archlinuxcn-keyring

#安装archlinuxcn-keyring

\$ rm -rf /etc/pacman.d/gnupg && pacman-key --init && pacman-key --populate archlinux && pacman-key --populate archlinuxcn

#生成新的密钥环并重新签署密钥 (安装archlinuxcn-keyring不报错时可跳过)

\$ pacman -Sy

#再次同步数据

\$ pacman -S mesa lib32-mesa xf86-video-amdgpu vulkan-radeon lib32-vulkan-radeon xf86-video-ati opengl-mesa opengl-headers

#安装AMD核显相关驱动 (不是AMD的可以跳过)

\$ pacman -S mesa lib32-mesa xf86-video-intel vulkan-intel lib32-vulkan-intel opengl-mesa opengl-headers

#安装intel核显相关驱动 (不是英特尔的可以跳过)

\$ pacman -S pipewire lib32-pipewire pipewire-media-session pipewire-alsa pipewire-pulse pipewire-jack lib32-pipewire-jack

#安装声音相关驱动

\$ systemctl enable bluetooth


```
#开机自启bluetooth服务（蓝牙服务，若无可以跳过）
```

```
$ reboot
```

```
#重启并登陆root
```

8.设置swap

在桌面环境中，交换分区或文件用来实现休眠(hibernate)的功能，即将当前环境保存在磁盘的交换文件或分区部分。除此之外，某些特定软件需要 swap 才可以正确运行。交换文件与分区性能相同，且交换文件更为灵活，可随时变更大小，增加与删除。

```
dd if=/dev/zero of=/swapfile bs=1M count=4096 status=progress #创建4G的交换空间 大小  
根据需要自定  
chmod 600 /swapfile #设置正确的权限  
mkswap /swapfile #格式化swap文件  
swapon /swapfile #启用swap文件
```

9.安装桌面环境

如果想要挑选自己喜欢的桌面环境的话可以参见[Desktop environments](#)推荐：KDE,cute fish

下面以KDE为例，其他的请自行某度

```
$ pacman -S ttf-dejavu ttf-liberation noto-fonts noto-fonts-cjk noto-fonts-emoji  
noto-fonts-emoji-blob noto-fonts-extra wqy-bitmapfont wqy-microhei wqy-microhei-  
lite wqy-zenhei ttf-arphic-extra ttf-arphic-ukai ttf-arphic-uming adobe-source-  
code-pro-fonts adobe-source-han-sans-jp-fonts adobe-source-han-sans-tw-fonts  
adobe-source-han-serif-jp-fonts adobe-source-han-serif-tw-fonts adobe-source-han-  
sans-cn-fonts adobe-source-han-sans-kr-fonts adobe-source-han-serif-cn-fonts  
adobe-source-han-serif-kr-fonts adobe-source-sans-fonts adobe-source-han-sans-hk-  
fonts adobe-source-han-sans-otc-fonts adobe-source-han-serif-hk-fonts adobe-  
source-han-serif-otc-fonts adobe-source-serif-fonts  
#安装字体（请根据需要自行补充，这里只安装常用的包）
```

```
$ pacman -S plasma-meta konsole dolphin kate ark gwenview vlc firefox  
#安装KDE桌面及软件（这里只安装最必要的包，如果想完整使用KDE的各种功能请根据对应提示安装需要的  
包）
```

```
$pacman -S sof-firmware alsa-firmware alsa-ucm-conf ntfs-3g adobe-source-han-  
serif-cn-fonts wqy-zenhei ttf-lxgw-wenkai noto-fonts-cjk noto-fonts-emoji noto-  
fonts-extra p7zip unrar unarchiver lzop lrzip packagekit-qt5 packagekit  
appstream-qt appstream gwenview git wget kate bind
```

```
$ systemctl enable sddm  
#设置开机自启动sddm服务
```

```
$ reboot  
#重启进入桌面环境
```

9.桌面中文环境的设置&输入法的安装及配置——fcitx5

1.中文环境设置：

System Settings（系统设置）>>Regional Settings（语言和区域设置）>>Language（语言）>>Add language（添加语言），找到**简体中文**后点Add（添加）。添加简体中文后，将其移到最上方，并删除其他多余语言，否则可能会出现汉化不全的情况。上述操作完成后，点击Apply（应用）。

System Settings（系统设置）>>Regional Settings（区域设置）>>Formats（格式）>>Region（区域），选择**简体中文（中国）**。上述操作完成后，点击Apply（应用）。

点击开始菜单，点击Switch users（切换用户），再重新登陆即可使用中文

2.Fcitx5

```
$sudo pacman -S fcitx5-im fcitx5-chinese-addons
#安装fcitx5主体、配置工具、输入法引擎及中文输入法模块

$ sudo vim /etc/environment
#编辑/etc/environment文件，使系统正确识别输入法，并在末尾加入以下内容：
GTK_IM_MODULE=fcitx
QT_IM_MODULE=fcitx
XMODIFIERS=@im=fcitx
#配置环境变量（注意大小写）

$ sudo pacman -S fcitx5-pinyin-zhwiki fcitx5-pinyin-moegirl
#安装词库

$ reboot
#重启
```

3.其他配置——

1.AUR helpers的安装：

有很多AUR助手，详情可参见Arch Wiki [AUR helpers](#)

```
$ sudo pacman -S paru
#安装paru

$ sudo vim /etc/paru.conf
#编辑/etc/paru.conf文件，删除BottomUp前面的#

paru -S 包名
#注：不能加sudo
```

2.其他软件源的添加

```
$ sudo pacman-key --recv-keys 7931B6D628C8D3BA && sudo pacman-key --finger
7931B6D628C8D3BA && sudo pacman-key --lsign-key 7931B6D628C8D3BA
#导入arch4edu源的GPG key

$ sudo vim /etc/pacman.conf
#编辑/etc/pacman.conf文件，在末尾加入以下内容：
[arch4edu]
Server = https://mirrors.tuna.tsinghua.edu.cn/arch4edu/$arch
```

```
#添加arch4edu源
[blackarch]
SigLevel = Never
Server = https://mirrors.ustc.edu.cn/blackarch/$repo/os/$arch
#添加blackarch源（暂时加入SigLevel = Never，否则会报错；注意大小写）

$ sudo pacman -Sy
#同步数据

$ sudo pacman -S arch4edu-keyring blackarch-keyring
#安装arch4edu-keyring & blackarch-keyring

$ sudo rm -rf /etc/pacman.d/gnupg && sudo pacman-key --init && sudo pacman-key --
populate archlinux && sudo pacman-key --populate archlinuxcn && sudo pacman-key -
-populate arch4edu && sudo pacman-key --populate blackarch
#生成新的密钥环并重新签署密钥

$ sudo vim /etc/pacman.conf
#编辑/etc/pacman.conf文件，删除[blackarch]区域的SigLevel = Never

$ sudo pacman -Sy
#再次同步数据
```

11.系统的更新

```
$ sudo pacman -Syu
#更新系统

$ paru
#更新系统及AUR软件
```

在更新时请先查看Arch Linux官网的新闻公告，看是否需要升级时人为的干预；切勿无脑更新！

END —(想再看一遍本教程吗？那就请在终端中输入`sudo rm -rf /*`，你会回来的。)—

Made in China

By Microsoft Xiaoice PILIHU
