



链滴

从现有 Linux 系统上安装 Archlinux[云环境 无 VNC]

作者: [hiplon](#)

原文链接: <https://ld246.com/article/1594189281844>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

以Oracle Cloud环境为例, 从现有Linux系统上安装Archlinux [理论上可以无VNC实现]

以Oracle Cloud环境为例,启用一个Oracle Linux 7.8的实例,opc登陆系统,sudo -i切换为root用户。
续后续操作:

原系统

```
[root@jpt2 tmp]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda   8:0    0 46.6G 0 disk
├─sda2 8:2    0   8G 0 part [SWAP]
├─sda3 8:3    0 38.4G 0 part /
└─sda1 8:1    0 200M 0 part /boot/efi
```

```
cd /tmp
wget https://mirrors.edge.kernel.org/archlinux/iso/latest/archlinux-bootstrap-2020.07.01-x86_64.tar.gz #下载最新archlinux-bootstrap压缩包
tar -xf archlinux-bootstrap-*.tar.gz

mount --bind /tmp/root.x86_64 /tmp/root.x86_64 #
vim /tmp/root.x86_64/etc/pacman.d/mirrorlist #
/tmp/root.x86_64/bin/arch-chroot /tmp/root.x86_64/ #第一层chroot命令
```

第一层Chroot

```
#第一层Chroot
export PS1='root@arch-chroot-1 #'
```

```
pacman-key --init
pacman-key --populate archlinux
pacman -Syy
```

```
mount /dev/sda3 /mnt
mount /dev/sda1 /mnt/boot/efi
```

```
root@arch-chroot-1 #lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda   8:0    0 46.6G 0 disk
├─sda1 8:1    0 200M 0 part /mnt/boot/efi
├─sda2 8:2    0   8G 0 part [SWAP]
└─sda3 8:3    0 38.4G 0 part /mnt
```

```
cd /mnt
```

```
#! ! ! ! ! 除了 boot、tmp、dev、proc、run、sys 几个目录外的其他所有文件 ! ! ! ! !
```

```
root@arch-chroot-1 #ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr
ar
root@arch-chroot-1 #rm -rf bin etc home lib lib64 opt root sbin srv usr var
```

```
newroot=/mnt
```

```
mkdir -m 0755 -p "$newroot"/var/{cache/pacman/pkg,lib/pacman,log} "$newroot"/{dev,run,etc}
mkdir -m 1777 -p "$newroot"/tmp
mkdir -m 0555 -p "$newroot"/{sys,proc}
mount --bind "$newroot" "$newroot"
mount -t proc /proc "$newroot/proc"
mount --rbind /sys "$newroot/sys"
mount --rbind /run "$newroot/run"
mount --rbind /dev "$newroot/dev"
```

```
pacman -r "$newroot" --cachedir="$newroot/var/cache/pacman/pkg" -Sy base linux linux-firmware openssh xfsprogs sudo vi vim
```

```
cp -a /etc/pacman.d/gnupg "$newroot/etc/pacman.d/"
cp -a /etc/pacman.d/mirrorlist "$newroot/etc/pacman.d/"
```

```
genfstab -U /mnt >> /mnt/etc/fstab
chroot "$newroot"
```

第二层Chroot #以快速配置为目标，不做其它额外配置

```
#第二层Chroot
export PS1='root@arch-chroot-2 #'
```

#检查一下/etc/fstab,看看uid,卷之类的有没有错误，我这边就重复生成了一个根目录的配置，要删除

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

```
vim /etc/locale.gen
#添加en_US.UTF-8 UTF-8
```

```
locale-gen
```

```
#新建systemd网络配置
vim /etc/systemd/network/20-wired.network
root@arch-chroot-2 #cat /etc/systemd/network/20-wired.network
[Match]
Name=ens3
```

```
[Network]
DHCP=ipv4
```

```
#配置root密码
passwd
```

```
#允许root 远程ssh登陆
vim /etc/ssh/sshd_config #添加PermitRootLogin yes
#添加PermitRootLogin yes
```

```
#启用DHCP网络和sshd
systemctl enable systemd-networkd
systemctl enable sshd
```

#以下引导内容视个人情况，理论上可以达到无VNC环境的覆盖安装原有的Linux系统

```
#编辑原来的grub配置增加archlinux启动项
grub-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

```
#我这边还要手动修改一下生成的grub配置文件
```

```
linux /vmlinuz-linux改成
linuxefi /vmlinuz-linux
```

```
initrd /initramfs-linux改成
initrdefi /initramfs-linux
```

接下来就可以爽快地玩Arch了

```
[root@archlinux ~]# neofetch
      _`      root@archlinux
     .o+`     -----
    `ooo/     OS: Arch Linux x86_64
   `+oooo:   Host: KVM/QEMU (Standard PC (i440FX + PIIX, 1996) pc-i440fx-2
9)          Kernel: 5.7.7-arch1-1
   `+ooooo:   Uptime: 6 mins
  -+oooooo+: Packages: 129 (pacman)
 `/:-:++oooo+: Shell: bash 5.0.17
  \+++++/+++++++: Resolution: 1024x768
  \+++++++/+++++++: Terminal: /dev/pts/0
  \+++ooooooooooooo/` CPU: AMD EPYC 7551 (2) @ 1.996GHz
 ./ooosssso++osssssso+` GPU: 00:02.0 Vendor 1234 Device 1111
 .oosssso-````/osssss+` Memory: 63MiB / 976MiB
 -osssssso. :ssssssso.
 :osssssso/  osssso+++.
 /osssssso/  +sssooo/-
 \osssso+/-  -:/+osssso+-
 `+sso+:-`   \-/+oso:
 `++:..     \-/+/
 `         \/
```

```
[root@archlinux ~]#
```

refer: <https://www.vnf.cc/2020/07/install-archlinux-from-linux/>