

Installing Arch Linux on a Virtualbox VM

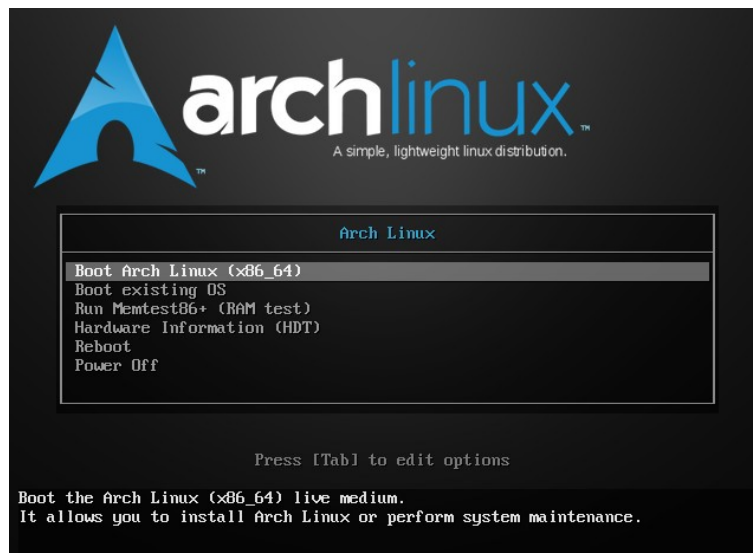
Using the New VM wizard

- Name and OS: *VMName* / Linux / Arch Linux (64-bit)
- Memory size: 2GB (*min*)
- Virtual hard disk, VDI, Dynamically allocated, 10GB (*min*)

Settings before first boot

- General -> Advanced -> Shared clipboard bidirectional
- System -> Processor -> 2 CPUs *if base system has 4+*
- System -> Acceleration -> Paravirtualisation Interface: KVM
- Display -> Screen -> 128MB video memory
- Display -> Screen -> Enable 3D Acceleration
- Storage -> Optical Drive: Arch install ISO
- Network -> Adapter 1 -> Attached to Bridged Adapter

On starting the New VM, the Arch install menu should display. Choose the first option.



After the system boots you are working as root to continue setup.

Set up for a UK keyboard during the installation

```
loadkeys uk
```

Prepare and mount the hard drive where Arch is to be installed. Example is just one partition with no swap partition. Use fdisk to allocate the entire drive as primary partition.

```
fdisk /dev/sda
```

```
n, p, 1, accept sector defaults, w
```

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

```
mount /dev/sda1 /mnt
```

Confirm that network is usable (ping a known host)

Optional: install your preferred console editor. (nano is provided)

```
pacman -Sy mc (installs Midnight Commander)
```

Edit `/etc/pacman.d/mirrorlist` and move UK mirrors to top of list

Install base Arch system to the mounted partition. (Takes several minutes)

```
pacstrap /mnt base
```

Create various configuration files that will be needed when the new system boots:

Generate fstab for new system

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

Edit `/mnt/etc/locale.gen` and enable `en_GB.UTF8` (uncomment the line)

Create `/mnt/etc/vconsole.conf` with the line

```
KEYMAP=uk
```

Create the `/mnt/etc/hostname` file and enter the computer name

Create the `/mnt/etc/hosts` file with minimum entries:

```
127.0.0.1 localhost  
::1 localhost  
127.0.1.1 compname.trosnant.home compname
```

Work with the mounted drive as the root filesystem

```
arch-chroot /mnt
```

Generate locale config

```
locale-gen
```

Set appropriate timezone and clock

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

```
hwclock --systohc
```

Create startup filesystem

```
mkinitcpio -p linux
```

Install grub bootloader and Intel microcode updater packages

```
pacman -S grub
```

```
pacman -S intel-ucode
```

Configure grub to boot new system

```
grub-install --target=i386-pc /dev/sda
```

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

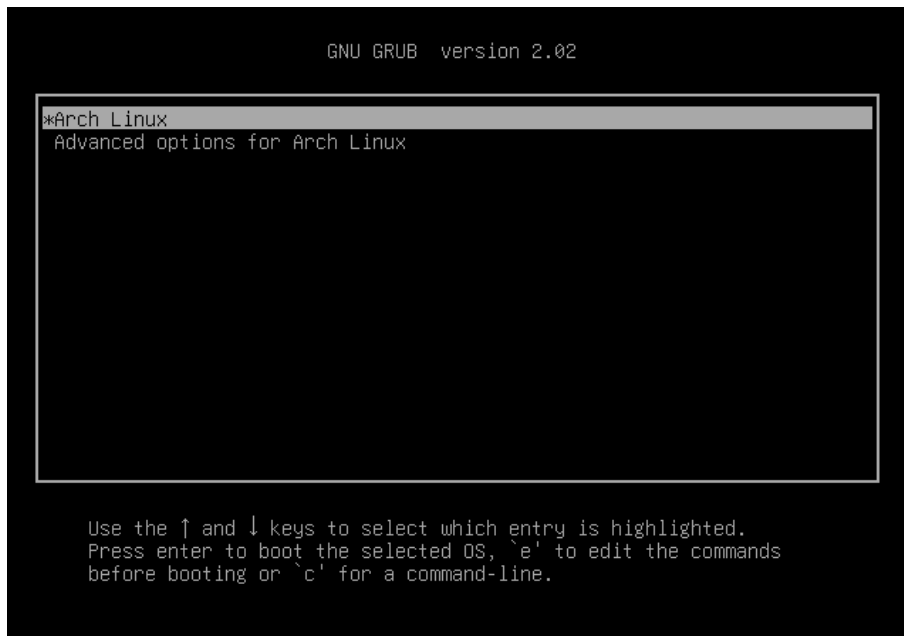
Halt the system ready for boot from the installed system

```
exit
```

```
halt -p
```

Important! Remove the Arch install ISO from the optical drive of the VM.

Start the VM. If all is well the new boot screen options should appear:



Log in as root to continue the installation.

Set networking to use DHCP to obtain IP config.

```
pacman -S dhcpcd
```

```
cp /etc/netctl/examples/ethernet-dhcp /etc/netctl
```

```
systemctl enable dhcpcd
```

Reboot and check network is now usable. (*Allow a short period for dhcp to obtain address and configure network.*)

You may want to take a snapshot at this point!

For a GUI, the first step is to get a working Xorg installation

See <http://wideaperture.net/blog/?p=4503>

```
pacman -Syu
```

```
pacman -S xorg-server xorg-xinit xorg-apps
```

```
pacman -S virtualbox-guest-modules-arch virtualbox-guest-utils
```

Optional: install your preferred console editor. (nano is provided)

```
pacman -Sy mc
```

Create the file /etc/modules-load.d/virtualbox.conf and add:

```
vboxguest
```

```
vboxsf
```

```
vboxvideo
```

Reboot to start the new configuration.

Enable the virtualbox services

```
systemctl enable vboxservice.service
```

Install some basic utilities for testing the Xorg server

```
pacman -S xorg-twm xorg-xclock xterm
```

You should now be able to start a basic X session

```
startx
```

You may want to take a snapshot at this point!

Install a desktop environment, eg Mate, and a graphical login manager (this will take several minutes)

```
pacman -S ttf-dejavu
```

```
pacman -S mate mate-extra
```

```
pacman -S lightdm lightdm-gtk-greeter
```

```
systemctl enable lightdm.service
```

Reboot to test

You may also want to create a non-root user.

```
useradd -m -g users -G wheel,storage,power -s /bin/bash user
```

```
passwd user
```



Copyright © 2018 Nick Talbott and Jim Price