My
Simple
Networking
Linux
Guide

Date: 15.05.2020

Contents

List of Figures

List of Tables

1 Introduction

In the following guide, you will learn how to set up, a simple connection between two linux machines.

2 Software Used

For this project the following external software are going to be used:

- Oracle VM VirtualBox Manager
- Arch Linux
- OpenSSH
- ...

This guide is not going to go into detail, on how to install the software "Oracle VM Virtual Box Manager". As the ability to install this is less than the minimum requirement to understand this guide.

3 Creating the Virtual Machines

After you open up "Oracle VM Virtual Box Manager", create two new virtual machines, for this you will need an Arch Linux ISO file, which you can download from the Arch Linux website. Once you have the ISO file, you can just press the Control and N key simultaneously or click the Star button. Here you will need to Name your virtual, locate and select the Arch Linux ISO file you downloaded and press continue. Afterwards you can leave everything on the default settings. The only specification that you will need to change is the RAM and in case you want to limit the storage space used, the storage space of your virtual machine, each of your virtual machines, should have at-least the following specifications:

- 2 Gigabytes of RAM
- 40 Gigabytes of storage space, in case of static storage

4 Installing the Arch Linux operation system

Disclaimer: This guide only goes into how to install Arch Linux on a normal BIOS, in case you wish to install it on a computer with "Unified Extensible Firmware Interface" or UEFI for short, the steps you will have to take are different! → you can check for this using the following command "ls /sys/firmware/efi/", if you have not found anything, your system booted up in a normal BIOS, otherwise it is UEFI.
After creating the virtual machines, you can boot up said machines. Once here you will be greeted by the following image Here just choose the "Boot Arch Linux (x86_64)" option, by pressing the enter key. After doing so, you will see Arch Linux booting up, do not be scared, just wait for it to do its thing, you will know it is done, when you see the following scene

4.1 Setting up partitions

Here you will first want to type in the command "lsblk", this will show you all the so called block devices on the virtual machine, the important one for us here is going to be the one, with the type disk and the same Size, as your virtual machines storage space. After checking that, you might want to check for an Internet connection, for this you can just use the "ping URL/IP" command for this, or in case you are connected using a wireless connection, use the command "wifi-menu" this will search for all the available wireless connections and allow you to access them, using a password of course, in case it is needed. Now, we can proceed to partitioning our drive, in order to begin you will need the "fdisk dev/yourDisksName" command. Here you can ask for help by typing in "m". You probably do not have any partitions, but in case you do make sure to delete them using the "d" command. Traditionally you have four partitions, these are the following:

- boot
- swap
- root
- home

Now the first partition, that you will have to make is the "boot" partition, "The /boot directory contains the kernel and ramdisk images as well as the bootloader configuration file and bootloader stages. It also stores data that is used before the kernel begins executing user-space programs. /boot is not required for normal system operation, but only during boot and kernel upgrades (when regenerating the initial ramdisk).", for this, type in "n" for new partition, first just press enter, as it is going to be a primary partition, then it will ask for a number, it is recommended to use the default "1" for this, so just press enter, after that, when it asks for the size of the "First Sector", just press enter again, now for the size of the "Last Sector" here we are going to use "+600M" or 600 Megabytes for this, but you can probably use less.

Next up is the “swap” partition, this partition is used like virtual RAM, you can do the same you did previously until it asks for the size of the “Last Sector”, here the size is recommended to be around 1.5 times the size of the machines RAM, in this case that is “+6G” or 6 Gigabytes, as the virtual machines I use have 4 Gigabytes of RAM each.
Now for the “root” partition, the steps are the same as previously until it asks for the size of the “Last Sector”, here at least around “+15G” or 15 Gigabytes should be assigned.
The last Partition you will want to create is the home partition, “The /home directory contains user-specific configuration files, caches, application data and media files.”, here first specify that it is a primary partition, then you can just press enter all the way through, as the rest of the available storage space should be assigned to this partition.
Now in order to actually make the changes you will have to type in the “w” command for write, this will wipe everything you had previously on the Storage Device, so beware.

4.2 Setting up file systems

After partitioning, you will wish to set up the necessary file systems, with the exception of the “swap” partition, every other partition is going to be made into a file system. You can do this using the “mkfs.ext4 /dev/nameOfPartition” command.

4.3 Setting up the swap partition

For this you just need to use the “mkswap /dev/nameOfSwapPartition” command first, this sets this partition up to be used as a swap partition, after which, you just have to tell the system to use it, using the “swapon /dev/nameOfSwapPartition” command. You can check, whether you succeeded by using the “swapon –show” command, this should show the partition, you labeled as a swap partition.

4.4 Mounting the partitions

Before a partition can be used, it needs to be mounted into a mount point.
For mounting partitions the following command can be used “mount /dev/nameOfPartition /destination_dir”. This does not apply to the “swap” partition.

4.4.1 Mounting the root partition

This is the first partition you will want to mount. Traditionally the “root” partition is mounted to the “/mnt” directory.

4.4.2 Making mount points for the other partitions

A mount point in this case is just a directory. You will want to make two directories inside the “/mnt” directory, you can make a directory using the “mkdir nameOfDir” command. The first one is going to be the home directory, you can either go into the “/mnt” directory using the “cd path” command or you can just use the path, in this case “/mnt/home”. The other one is the “boot” directory.

4.4.3 Mounting the other two partitions

You can simply use the same command you used to mount the “root” partition, so “mount /dev/nameOfPartition”.
/ After doing so you should type in the “lsblk” command, and be greeted by the following, see picture

4.5 Actually setting up Arch Linux

Now you can finally, install Arch Linux, the command you will need for this is “pacstrap /mnt base base-devel linux linux-firmware vim”, “/mnt” here is your root directory, after that comes all the packages you wish to install. We install the base packages, the base packages for developers, linux, the linux firmware, and last but not least, VIM of course. Why VIM, cause why not? In case it asks you to choose an option, always go with the default.

4.6 Making an fstab file

In order to avoid, having to manually, mount all of our partitions, every time, we boot up the machine, we will need an fstab file. This file contains information on how devices should be mounted into the file system. We could create one such file manually, but as Arch Linux provides a command that does it automatically, we will use that, which is the “genfstab” command. Now on its own the “genfstab” command only generates an unreliable table, so we will use the following “genfstab -U /yourRootDir ¿¿ /yourRootDir/etc/fstab”, this command will generate a more reliable version and write it into the “fstab” file inside the “etc”, which we specified to be in your root directory.

4.7 Making Arch Linux bootable

For this you will need to change the root directorys location using the “arch-chroot /yourRootDirectory” command.

4.8 Installing a network manager

By default, Arch Linux will not detect and setup your internet access like windows does. So for this we are going to use pacman, the package manager that came with our packages. Just type in the following command “pacman -S networkmanager” and press enter, pacman will automatically install the network manager for you. You will naturally have to confirm the installation. But as you will need the network manager to automatically start upon loging in, you will have to use the following command “systemctl enable NetworkManager” to tell the system to do so.

4.9 Installing a boot loader

You will need this aswell, so that the system can start up all the tasks needed, when booting, such as loading the operating system into memory. We will be using “grub” or GRand Unified Bootloader for this. Just type in “pacman -S grub”, press enter and let the package manager handle the rest. You will need to confirm the Installation of course.

“Setting up grub” Sadly “grub” does not get automatically set up, by the package manager, so we will have to install and configure it. First you will need to install it using the following command “grub-install –target=i386-pc /dev/nameOfDisk”, you will have to use the name of the disk and not the partition! After installing grub, we need to create a configuration file, for this you can type in “grub-mkconfig -o /boot/grub/grub.cfg” and press enter, this will create the configuration file for grub, named “grub.cfg”, which is located und boot/grub in your root directory.

4.10 Setting the hostname

For this just use your text editor, we use vim, and open up the following file “/etc/hostname”, type in your desired hostname and save it.

4.11 Setting the Password

There is supposedly a default password “root”, but that sometimes decides not to work, so it is a good idea to set the password of the “root” using the “passwd” command.

“Other things” A few things you might want to setup, but this guide does not go into are the following:

- Time-Zone
- Languages

5 Setting up a Secure Shell using Open SSH

5.1 Installing the package

Now we could have installed the package together with Arch Linux, but we will do it here. To install “openssh” just use the package manager, here is the command “pacman -S openssh”.

5.2 Setting the Port

In order to set the port, which the secure shell client is going to use, you will need to open the “sshd_config” file in your preferred editor, which is located in the “/etc/ssh/” directory. We use vim, so the command will be “vim /etc/ssh/sshd_config”. Here you will want to remove the “” sign before the “Port” and change the default value, which

should be “22” to your desired Ports number.
The Ports we are going to use are the following:

- Client 1: 123
- CLient 2: 124

6 Setting up a virtual private network using Open VPN

7 Service

8 Setting up the firewall