Justice Escalante

1/31/23

**How to install Linux Distributions/Flavors; Ubuntu**

Ubuntu

**Set up/Create your VM**

1. To create a VM you must first download one of the respective platforms; VMware, Azure, Virtual Box etc

Here are a few links to get you set up:

* 1. <https://www.virtualbox.org>
  2. <https://azure.microsoft.com/en-us/free/search/>
  3. <https://www.vmware.com>

1. Proceed with the installer found on the web
2. Once done installing, your platform should look something like this if you are using VMware. *Image provided below for better clarity*

Graphical user interface, application

Description automatically generated

**Download the OS**

1. Now that you have a platform setup, it’s time to find your flavor. In this case we will be using Ubuntu. It can be found here: <https://ubuntu.com/download/server>
2. Once downloaded, go to your respective downloads folder, and keep an eye on where the iso is installed. Now open your VM application and navigate to the home page and start creating a new VM. If using Virtual Box or any other platform, it might look a bit different. *Image provided below for better clarity*

Graphical user interface, application

Description automatically generated

1. Now once you click *Create a New Virtual Machine* it will bring you to the initial configuration page, in this case I have picked typical. Custom as an option is also viable if those said settings are needed: usually more in depth stuff will be located here. This will then bring you to where an iso file as a OS input is needed. *Images provided below for better clarity*

Graphical user interface

Description automatically generatedGraphical user interface, text, application

Description automatically generated

1. After picking the OS and installing the iso, you should be able to boot up your VM an start the actual installer. On boot most server distros will be keyboard only for input, so keep that in mind when setting up a new OS.

Text

Description automatically generated

**Set up your respective settings during install.**

1. Setting up your OS properly during any install is very important as there might be things you want to include such as openssh which is used for various It things such as remote connection and overall server management. Openssh also allows for scp which is highly important to share files quickly and securely. Other settings like scp or various Linux tools will be here etc. *Images provided below for better clarity*
2. Some other important settings to look out for when installing this distro is stuff like configuring your proxy, hardware settings etc. Configuring if your server uses a proxy or not is important as proxies serve as sort of a mask when it comes to your ip. This is key to people with companies or people who want to stay anonymous better. Proxys are also tied to a region, so if you wanted a server for EU content, you could make one. Hardware settings for your vm/distro are super important as you can cause hardware limits and throttles if not set up properly. Hardware such as your CPU, Memory and Storage can pretty much halt your VM if its not allocated properly. For example, if you gave the VM 500mb of memory it will have serious throttling under load as not a lot of data can be temporarily stored there. If you need to change hardware settings after the fact, VM has a menu to do that which can be found in the home menu. Visuals for this below

Graphical user interface, text, application

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated

Text

Description automatically generatedText

Description automatically generated

1. In this distro network connection was already enabled so that was covered for us. But if it’s not you can edit the network interface with ***ifconfig*** and a few other tools. More on how do to this here: <https://linuxhint.com/ubuntu-enable-and-disable-the-network-interface/>

**Sign in**

1. When your OS is fully installed and the VM is powered up, you should see a terminal on boot up. Sign in with your respective details that you put in the installer. Keep in mind that when you’re typing your password, it won’t display stars like most logins. It will stay blank even when entering chars so don’t panic. *Image provided below for better clarity.*
2. *Now if you forgot your sign information, you could sign in as the root user and change it with the command* ***passwd <username>.***

Text

Description automatically generated with low confidence

**Make sure your connected.**

1. There are a few ways you can test to see if you are connected properly to the internet, one easy and preinstalled feature you could use with Linux is ping.
2. If the network is setup correctly ping should work verifying you set it up properly, if not ping should not work.
3. Here is a great example of how ping is used, the command in the screenshot below is***ping -c 4 google.com*** *Image provided below for better clarity*

Text

Description automatically generated with medium confidence

**Integrate a file sharing service**

1. There are a few file sharing services you can use for Linux, SCP, SmarTY, Putty etc, but use what’s best for you. In this case SCP was installed using the installer so that was nice! But if you need to, use the terminal to install like CentOS. If that was case SCP would be installed using the command: ***sudo apt install -y openssh-client openssh-server*** on Ubuntu *Image provided below for better clarity*

Text

Description automatically generated

1. To validate that SCP is working properly, you must first find your ip to connect using a command in the terminal. The command in use would be ***ifconfig***: it should look something like this if done properly, with your respective properties listed.
2. The information you will be looking for when this command is used is your inet ip, which in this case is displayed as *192.168.253.136.* To shoot files over to this server that ip would be needed. There are many ways to use the SCP command as well, refer to <https://linuxize.com/post/how-to-use-scp-command-to-securely-transfer-files/> for more information. *Image provided below for better clarity*

Text

Description automatically generated

1. In the images provided below I have used the ip from the server I have set up with CentOS, the ip was pulled using the same command: the ip in this case is *192.168.253.135*
2. I have also used the command *touch testingSCP2* on myUbuntu VMjust to make a file that could be transferred for documentation purposes that can be seen below.
3. Now you can test to see if the SCP command and everything you have done up to this point has worked! In the image below you can see the command in the terminal as *scp testingSCP2 192.168.253.135:/home/jescalante* has been executed and its result.
4. In the second screen shot is the CentOS server running commands such as *ls* that have properties to show what’s in your current directory *Images provided below for better clarity*

Text

Description automatically generated



1. In the screenshot above you can see SCP executed correctly as the file *testingSCP2* was transferred from this Ubuntu Vm to the CentOS one!