How_to_Create_a_Network_of_Machines_in_VirtualBox_with_SSH_A

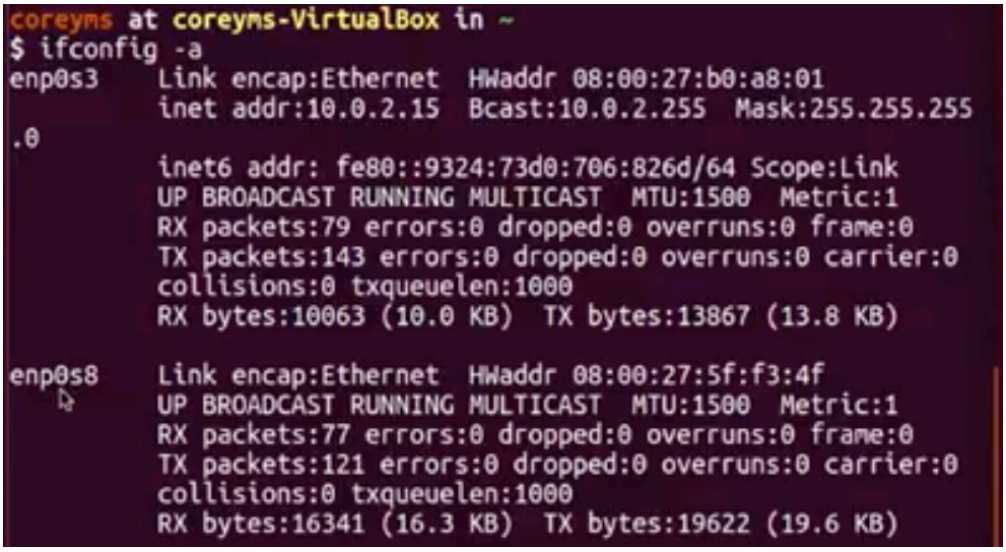
January 30, 2018

1 How to Create a Network of Machines in VirtualBox with SSH Access

Video Tutorial by: Corey Schafer

Instructions

- Go to File > Host Network Manager, and click Create.
- Go to DHCP Setting, and Disable them. Were giving our machines static IPs.
- Select the Ubuntu Desktop you have installed, and click settings.
- Naviagate to the Network Tab, check out the settings for adapter 1. 'Attached to:' should be set to NAT, and it should be enabled.
  - This will allow our guests to make an internet connection through our host connection.
- Go to adapter 2, enable it, and change 'Attached to:' to Host-only Adapter. Be sure its Name is set to the same one we just created in the Host Network Manager
- Go to the Base Ubuntu Desktop you created, Ctrl click and select Clone.
- Name the clone 'ubuntu-1', and click the checkbox that reinitalizes the MAC address of all network cards.
- Select Full Clone
- Select Current Machine State, and click clone.s
- Repeat the process, but change the clone's name to 'ubuntu-2'.
- Launch both Clones and log in using your original credentials from the base machine you cloned from.
- Open terminal in both, and check to make sure they have an internet connection. You can do this by typing `ping google.com`.
- Now we need to change the hostnames of both machines, because they are identical, which is not what we want.
- BOTH MACHINES: Do this by typing `sudo nano /etc/hostname`.
  - Remove both hostnames currently in the files and change one to ubuntu-1, and the other to ubuntu-2.
  - Type Ctrl-X, y, and press enter.
- BOTH MACHINES: And typing `sudo nano /etc/hosts`.
  - Change the name under localhost to the name used in /etc/hostname.



ubuntu-1

  - Type Ctrl-X, y, and press enter.

- BOTH MACHINES: Type `ifconfig -a`.
  - The `inet addr:` is bound to our first adapter
  - We are going to set enp0s8's `inet addr:` to the ip address in our Host Network Manager

- BOTH MACHINES: Type `sudo nano /etc/network/interfaces`
  - Add the following, BUT CHANGE the 1 to a 2 for ubuntu-2:

```
auto enp0s8
iface enp0s8 inet static
address 192.168.56.1
netmask 255.255.255.0
```

  - This sets up a static ip for the machine.
  - Type Ctrl-X, y (save), and press enter.
- BOTH MACHINES: Run `sudo reboot`
  - Those changes shouldn't need a reboot to take affect, but better safe then sorry.
- BOTH MACHINES: Log Back in, and open terminal.
  - Should see that the hostname has changed.
- BOTH MACHINES: Re-run the `ifconfig -a` command.
  - The static ip address you assigned to enp0s8 should now show up.
- BOTH MACHINES: Now have either machine ping the other to test that they are indeed on the same network.

SSH-fail

```
$ ping 192.168.56.<other_machines_number>
```

- Now lets check if our host machine has access to both. Open terminal, and ping either machine.
- Now lets check if we can ssh into either machine.

```
ssh <username>@<machine's IP>

* Now this might happen:

Lawerences-MacBook-Pro:~ lawerencelee$ ssh lawerencelee@192.168.56.1
ssh: connect to host 192.168.56.1 port 22: Connection refused

* This means you probably do not have Open SSH installed, a SSH server.
* BOTH MACHINES: Run `sudo apt-get update` and `sudo apt-get install openssh-server -y`
* If you did happen to have it installed and were still not able to connect try restarting the
`sudo service ssh restart`.
```

- Now try shh-ing in again.
- Once ssh-ed in, try `uname -a` to verify details about the machines OS (i.e. OS version, date it was initalized, etc.).
- Should you encounter the following error:

This can occur if you've been messing around with a few different machines, and you get to the point that you need to remove your keys belonging to a hostname from a know hostfile. * To remedy this type the following:

```
$ ssh-keygen -R <ip address of machine>
```

This removes the old key and stores it in a file, and then generates a new one. * Now have VirtualBox create snapshots for either machine (TURN OFF MACHINES FIRST). * Set their names to SSH Setup, and place their static IPs in the description.

Lets make it so we can ssh in to our machines using their hostname. * On your local machine, type `sudo nano /etc/hosts`. * In the file add:

```
192.168.56.3    ubuntu-1
192.168.56.2    ubuntu-2
```

- Now we should be able to ssh in like so:

```
$ ssh <usr name>@<hostname>
```