P4-BMV2-Mininet VM Setup Instructions

CS6220 Topics in Networks

Objective: We need to set up an Ubuntu Virtual Machine on a host machine that already has the necessary software SDK installed to run P4 based switches on a mininet environment. It is also required to test the correct working of the same before attending the hands-on session for a more productive time.

Please follow the steps to get the system set up on your machine:

1. Installation of Hypervisor

We need to first install virtualbox(or an equivalent hypervisor) on our system. Download the binary from the links below according to your host OS:

Windows: https://www.virtualbox.org/wiki/Downloads
Linux: https://www.virtualbox.org/wiki/Linux_Downloads

The installation should be straightforward. Stick with the default settings.

2. Download the VM image

The link below contains the list of VM releases (.ova files) to choose from.

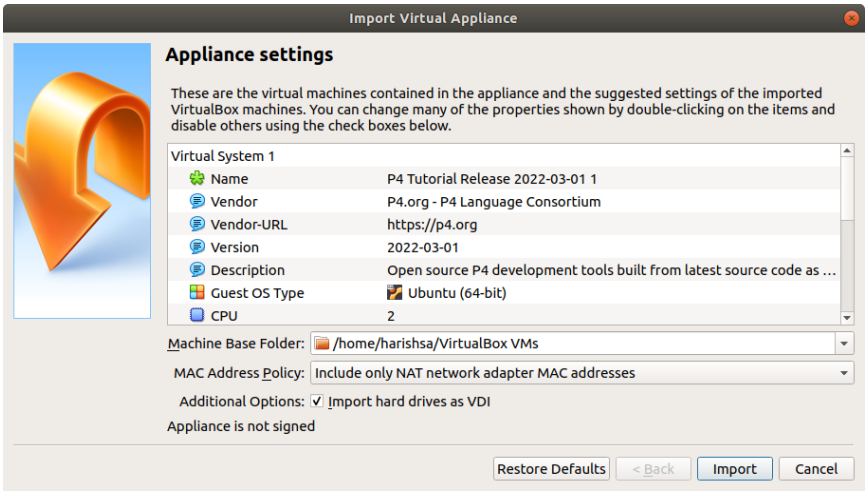
https://github.com/jafingerhut/p4-guide/blob/master/bin/README-install-troubleshooting.md

We can proceed with the latest release on 2023-Sep-01. Download “Release VM” and store it on your hard disk (You can use the below link to download the VM):

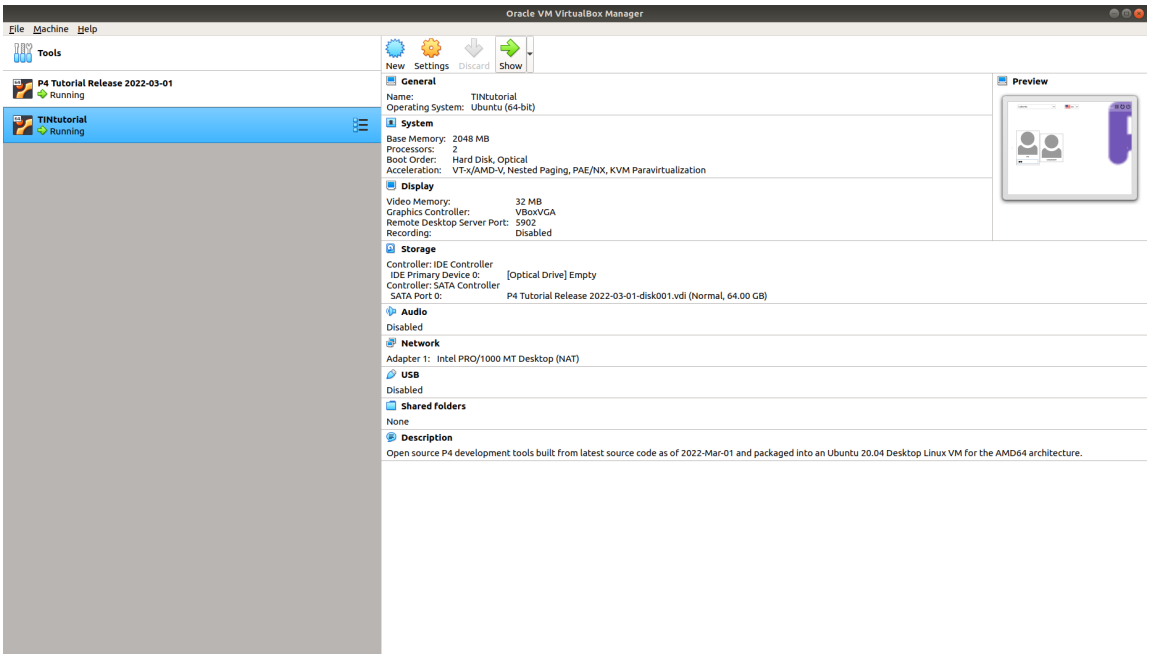
https://drive.google.com/file/d/1O28M8OdP_uq8tvDIbHaW1Zl1RqNT0XU3/view?usp=sharing

3. Import the downloaded VM image

Navigate to the installed location and double click on the .ova file you just downloaded. The screen below would be visible. Click on import and wait for the process to finish. You can change the name of the VM by double clicking on the name attribute. For the rest of the guide, the VM has been renamed to “TINTutorial”

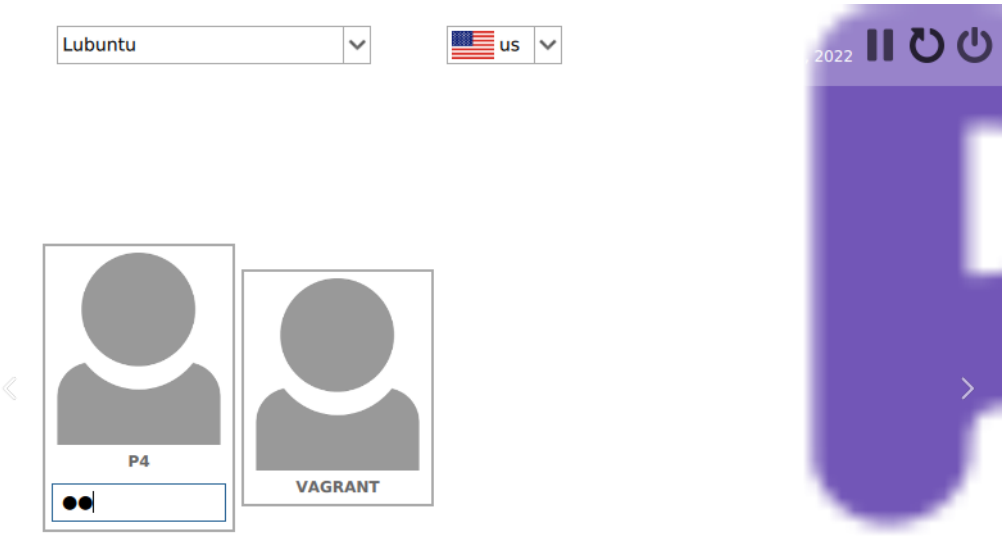


The imported VM ““CN Tutorial”” would be visible on the left pane as shown. Click on it and click start on the right pane.



Error Possibility: If there is an error with phrases like “VT-X” or “Virtualization not enabled”, it is most likely that Virtualization technology is not enabled on your system BIOS settings. Please boot into your BIOS settings and enable the same. The instructions are specific to the make of a system. A google search on how to enable virtualization can fetch them.

Assuming all goes well, choose the user P4. The password is “p4”.



Since this VM is minimal, one can install a browser(if needed) with the following commands on a terminal (can be opened from desktop):

> sudo apt update
> sudo apt install firefox

Type ‘y’ for any prompts that may appear

Now the VM is ready. Next we verify the environment

4. P4 Mininet Environment Verification

Open a terminal and navigate to the following location:

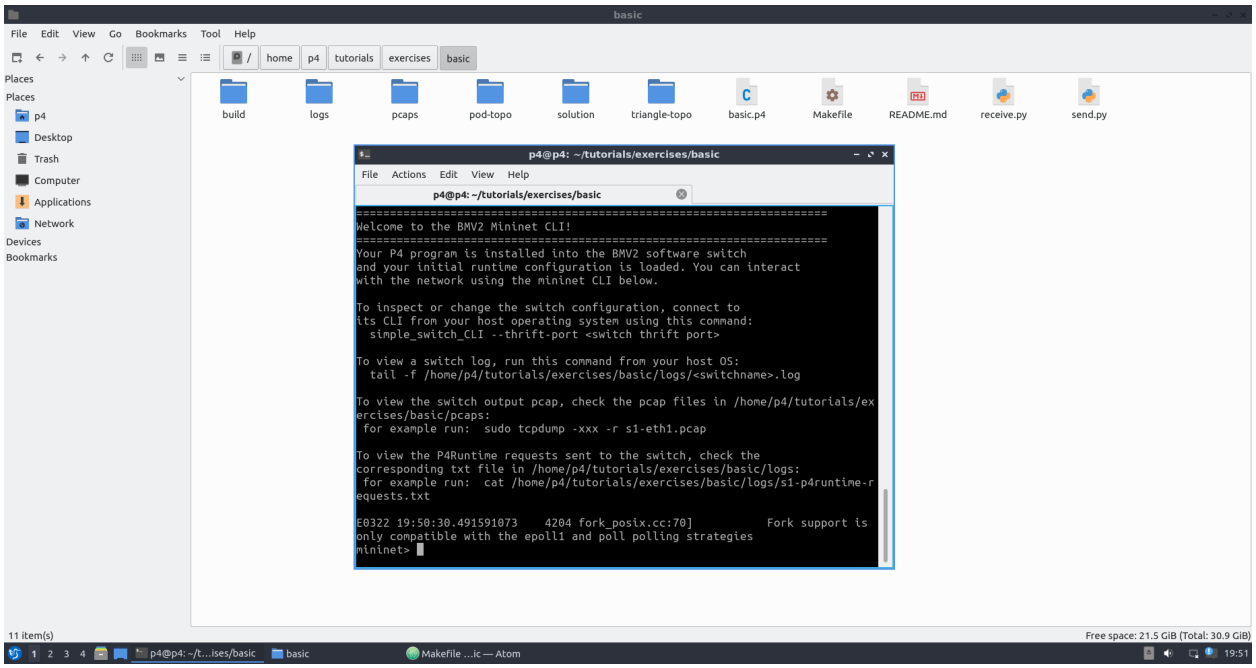
/home/p4/tutorials/exercises/basic

Can be done by running the command “cd /home/p4/tutorials/exercises/basic” on the terminal.

Once inside the location execute the following command on the terminal:

> make

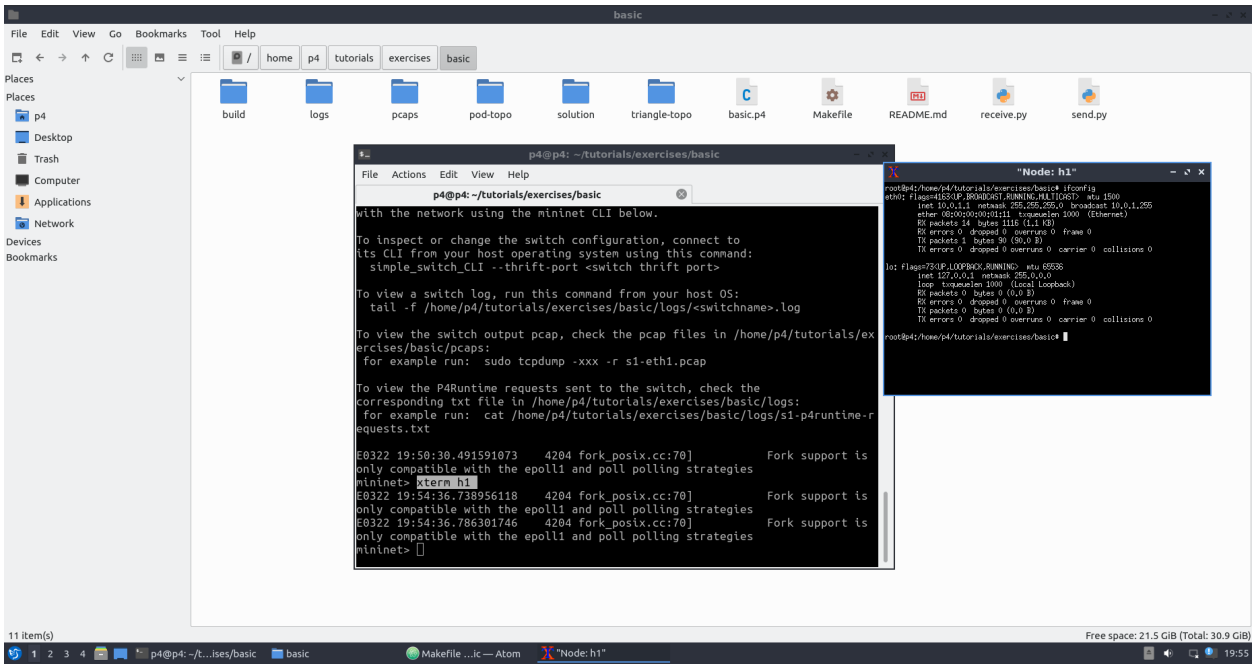
The screen shown below should be visible. We have now successfully started a mininet topology containing P4 compatible switches, hosts and links.



Next, inside the mininet prompt type the following command:

> xterm h1

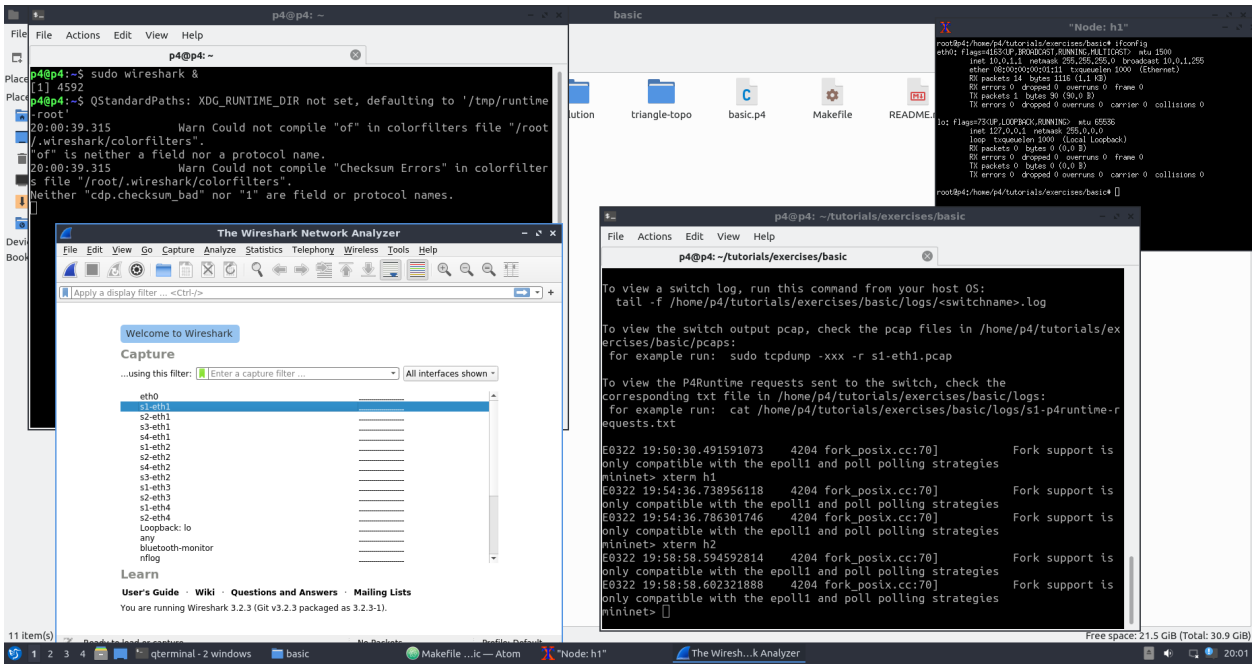
A new xterm window as shown below should open up. Type “ifconfig” in the prompt and verify if the screenshot and your result are the same.



Now we will generate packets and ensure that we can capture them on wireshark. So open up a new terminal and type the following command:

> sudo wireshark &

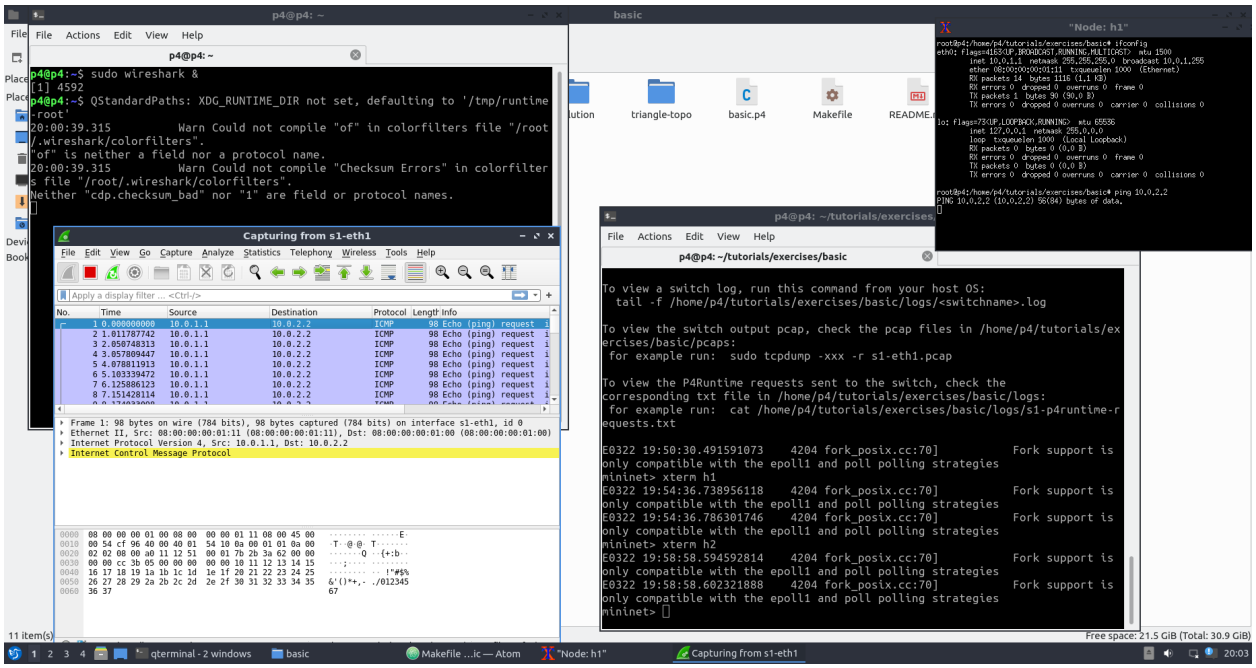
Once wireshark opens up, double click the interface “s1-eth1” as shown:



Now go back to the xterm window and type the command:

> ping 10.0.2.2

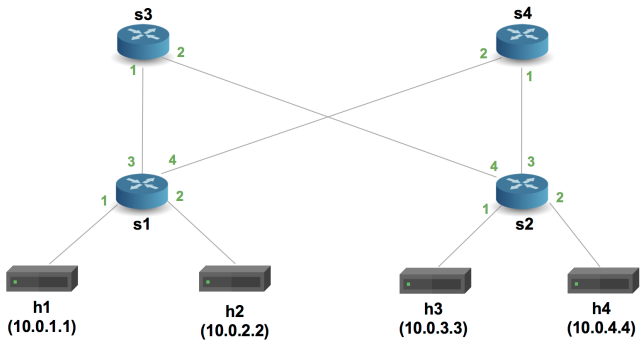
The wireshark screen will show ICMP packets captured



If all of the above has been achieved, then you have an environment ready to code P4 programs and process packets the way you want!

Extra Information (Optional Read)

We have basically simulated an environment containing switches, and hosts connected to them. In this specific case we have 4 switches and 4 hosts connected like this:



We achieved this when we executed “make run” in the terminal.

When we executed “xterm h1”, we got a terminal access to the host h1 with IP address 10.0.1.1 which we verified using “ifconfig”

When we opened wireshark, we were listening to the interface link between h1 and s1 (s1-eth1).

When we executed “ping 10.0.2.2” from the xterm window, we tried to ping h2 from h1 whose ICMP packets could be seen on the wireshark GUI.