# 4.2t:transcript

Hi and welcome to this week's tutorial.  In this one we are actually going to deploy a mininet within the VirtualBox environment that we've been using and creating over the past few weeks so we're going to create a simple virtual SDN network using a python-based POX controller built into mininet so the control is limited in functionality.  However, it's suited for the purpose of this tutorial so we're going to download a pre-configured mininet virtual machine and we're going to import that into VirtualBox.

As you can see on screen, I'm already at the web page but what I've actually done here, I've just put the link in notepad just so you can read this properly because it's not exactly a straightforward simple URL, it’s a little bit long so you want to go to [https://github.com/mininet/mininet/releases (Links to an external site.)](https://github.com/mininet/mininet/releases)

Okay, so if you write that down you can actually just go to the website fairly quickly.  Okay, I'll close that get it out of the way, and No, I don't want to save that okay!

So the page itself, when we go to it, this is what you'll see meaning that 2.22 is the current version so we are going to use the current version.

If you're watching this at some point in the future obviously, go for the latest version.

When we scroll down a little bit the one we're actually looking for is server AMD64.zip at the end that one there, so just make sure you select that one when you do.  It is a zip file so once we've downloaded this we're going to need to extract it as well but we'll talk about that when we get to the extraction point.

You can also see at the side here its 562 Megabytes.

It's obviously not a small file so it's going to take some time to download and it will take some disk space so I'm going to start the download now and then I'm going to hit pause on the video.

We will come back once it's downloaded.

Okay, so hopefully your file has downloaded, mine has and there is the zip file there, I've actually already extracted mine, it's a real simple process to do, just use whatever extraction system you actually have and you'll get this file here, in this folder.

If we open this folder and dig into it you can actually see the mininet package is deployed in what is known as an OVF format which stands for open virtualization format.  OVF files contain instructions for different virtualization platforms to deploy the embedded virtual machine within the OVF file, therefore it’s standardized and usable on many different platforms without having to create a new environment for each virtualization platform so in this folder you'll see these two files, one is the OVA file which should look like a little golden box, you can just see it up there now, if you're using VirtualBox that is, and the other one is really information to help run the mininet environment.

So to start this, all we actually do, is we double-click on the little gold box file. It will then, as you can see, open up to import into our VirtualBox, you can see there's our ones from the previous weeks, but this one has all the settings done for us so, you can see if you scroll down there, all the different bits pieces set, so in reality, all we need to do is basically hit the import key, and that it will start to import.  I'm going to pause this because it can take a little bit of time and we'll come back when we're ready to login.

Okay, and there we go, there's my mininet VM sat there, ready to go powered off so all we do like any other virtual machine in VirtualBox we're going to start it up, so again, depending on your machine, this can take a little bit of time, it's just confirming my machine is the one I want to run.  I'm going to click yes obviously, I'm going to pause this one at boots and we'll come back one when we're ready to login.  Ok, so as you can see my mininet has booted and got ready to run, as you saw just a few seconds ago while I was booting up you might get your machine asking for certain permissions to run, just click ok to those, it's absolutely normal, I know we haven't had that in previous videos but occasionally it happens if Windows needs to do an update or something like that.  Anyway, so once we are at the login prompt, we will log in and the username and password are both mininet and the password of mininet.  Okay, and we're in, so you can see there it's actually asking do I want to update, but I'm not going to bother updating it at the moment because, for our purposes we're just going to keep it really really straightforward and simple, so what we're going to do now is actually we want to load a mininet environment and it's a default basic network of one controller, one open V switch and two hosts connected to a switch so all you do is type MN and hit enter.  My machine is asking me to do as root so as we've done in previous weeks, sudo mn (super user mn) there we go, and it actually runs as root, that's absolutely fine.

So mininet tells you what the topology layout is and how each link to every device on the network is connected.  This can be simplistic or as complicated as you like, so we can do a simple check on whether the network is operational by doing pings, say from Host 1 to Host 2, you can see they're marked as H1, H2 so if you just look down here you can actually see it's adding the hosts there, adding the switches, adding the links H1 to S1, H2 to S1 and it's configuring the hosts, H1, H2 and it starts the controller, and it starts the switch. So if we actually just do a very simple h1 to ping h2, we actually get a pink coming back, ICMP, we talked about this a few weeks ago didn't we, so that will actually continue just to ping over and over again, so what's actually happening here, is that, as we're doing the ping, it's using the associative network topology, as a means to communicate.  So in this, a new flow table will be set up with the SDN controller and the ping will be sent across the open V switch so that will go on for as long as we wanted.

To stop it, press keys ‘control’ and ‘C’, that breaks us out of it, so we can also have this process carried out automatically by typing a command called ping all.  So, if we do that now, type ping all and hit Enter and now you can see it says, very quick, h1 to h2, h2 to h1, results 0% dropped, 2 of 2 received.  Obviously, if this was a bigger Network it would actually take a little bit more time it would be far bigger.

So to come out of the mininet we have, if we type exit, we're back at our command prompt. So we can actually see from this, what we are actually running is we're running Python scripts, and we can actually see what these scripts actually are, if I change directory to mininet examples, so you can actually see now, I'm in the mininet examples directory, and if I do ‘ls – l’.  For those of you who are not unix/linux familiar, LS minus L, allows me to list what's actually in that particular folder, and giving it to me in a list format.

If I just did ‘LS’ it wouldn't give me all the extra information here you can actually see in white, which is the owner and the user and all those types of things, so we can actually see these files are all finishing pretty much with dot py.  That actually means they are Python scripts so we could take a look at one of them, that one there, for example, treeping64.py.

Now in order to do this we actually need to use an editor called Nano, but I think, just before we do that, I think might be an idea just to run it and actually see what this actually does.  So in order to do so,  type sudo again, dot slash, that in UNIX basically tells me I want to run it from this directory so Super User do, from this directory and we're going to run treeping64 dot py.

You see that starts to build up this rather large system, it also automatically does a great deal of pings, so what I'm going to do is, I'm just going to pause while that finishes, rather than interrupt it, which I could with ctrl C if I wanted to, but I'm going to leave it run, make sure it's all healthy. Obviously, so for when you do it, you'll know it's all healthy, and we'll come back once it's done.

Okay, welcome back, that completes successfully for me and hopefully, it has for you as well.  So what you actually saw there while that was doing it, it was doing iterative pings, every device was pinging every other device, the Python script was set to do that, and it was not a large network by any means but of a reasonable size.  If like I said, just before I pause the video, if you want to break out of that it does work but ‘ctrl C’ will actually get you out of that script running and take you back to the prompt.  It won't do any harm so if it's still running if your machine is not quite that quick, and you want to move on, ‘ctrl C’ will work absolutely fine for you.

So we can actually have a look now at that file we have just run, and I briefly mentioned this idea of an editor, it’s called Nano, or at least one of them is, as there's lots of them.   So what I want to do is type the words Nano treeping dot py and what Nano allows me to do is edit the script, and I can start to look into the script and so there are several variables that control the outcome of the network, so you'll actually see a variable called depth, there it is there, so you can see network equals TreeNet, depth equals two, so our depth terms the amount of layers of switches in the topology, before we can connect back to the main controller so the fanout variable, the next one to it, determines how many switches are connected to a common switch at the end of the network.  And the number of hosts is using the fanout part, sorry the depth is the number of switches fanout part, is that the number of hosts.  So you have to be careful when you're playing about with these because there are actually exponents so in this case it's eight to the two so we have 64 hosts, so we need to be very careful when we're actually doing this because if you create some large figures in here you can very quickly create some very large networks.  So what I'd like you to do at some point is actually have a play with the depth and the fanout.   Like I say, if you create something that's extremely large you'll either start to slow your machine down badly, or it could potentially crash, but that is the beauty of having a virtual machine as well, it's not going to do is any harm at all, or if your script is running for a long time we can just use ‘Ctrl C’ and break out of it as well.  So have an experiment with the different network sizes changing the depth of fanout variables so like I say, be careful not to make them too big. So, if you ever have any issues with mininet, for example, you quit before the application ports finish reading the topology, you may discover some unwanted behaviour.  We'll just talk about that briefly before we finish.  If you're using nano, the controls down here, the little hat, that they're, so you've got the Hat and the X, that stands for control, so control X will exit, control O will write out, control R will read and so on.  So, I'm going to exit so I'll just do control X and it takes me back to the actual prompt.  So, as I was just saying, you can actually get various issues with mininet and there's easy ways to actually resolve it, and one of them is actually putting the minus C switch at the end of MN.  So, if we were to do sudo MN minus C.  I'm actually in the wrong area for that but the minus C switch when you're in the right area will actually clear the mininet down for you as well and set everything back to its defaults.

So, that's really it, that's what you're going to play around with today.  Have a look at that depth and that fanout and try some various different values and you'll see things differently, maybe go for a slightly larger depth and a smaller fanout and see what actually happens.  It will automatically still do those pings and so on.  So, finally, the only thing left to do is obviously shut our virtual machine down.  There’s a variety of ways we can actually do this, we can do it from the VirtualBox manager itself, or we can actually issue various shutdown commands.  So, if I just exit back to the login that will also allow me to do it and then you can just shut the Machine down as from over here, we can just stop it and it will close down.

Okay, so, have go with mininet, enjoy it.  Have a play around with that Python script and if you're feeling really adventurous, try and play about with some of the others in there as well.

Thanks and I'll see you next time.