CS312 :: Lab Week 6 :: VM Networking

In this lab, we'll be setting up a three-router, two computer virtual network, and then configuring a Virtual Private Network on top of it, complete with firewall rules.

Supplies needed

- Personal laptop with VirtualBox installed
- The VM Appliance archive from our Canvas website (which unpacks into 5 virtual machines) called "Lab5_RoutersAndCompys.ova", *and* the "CentOS_GUI_Reference.ova" appliance.
- The credentials.txt file that lists all of the usernames and passwords.

Procedure

Please follow these steps to complete the Lab. When asked, make sure you gather the information for the Questions as you go.

1. Import all 6 VMs into VirtualBox on your laptop by clicking the File -> Import Appliance for each .ova file, and following the steps.

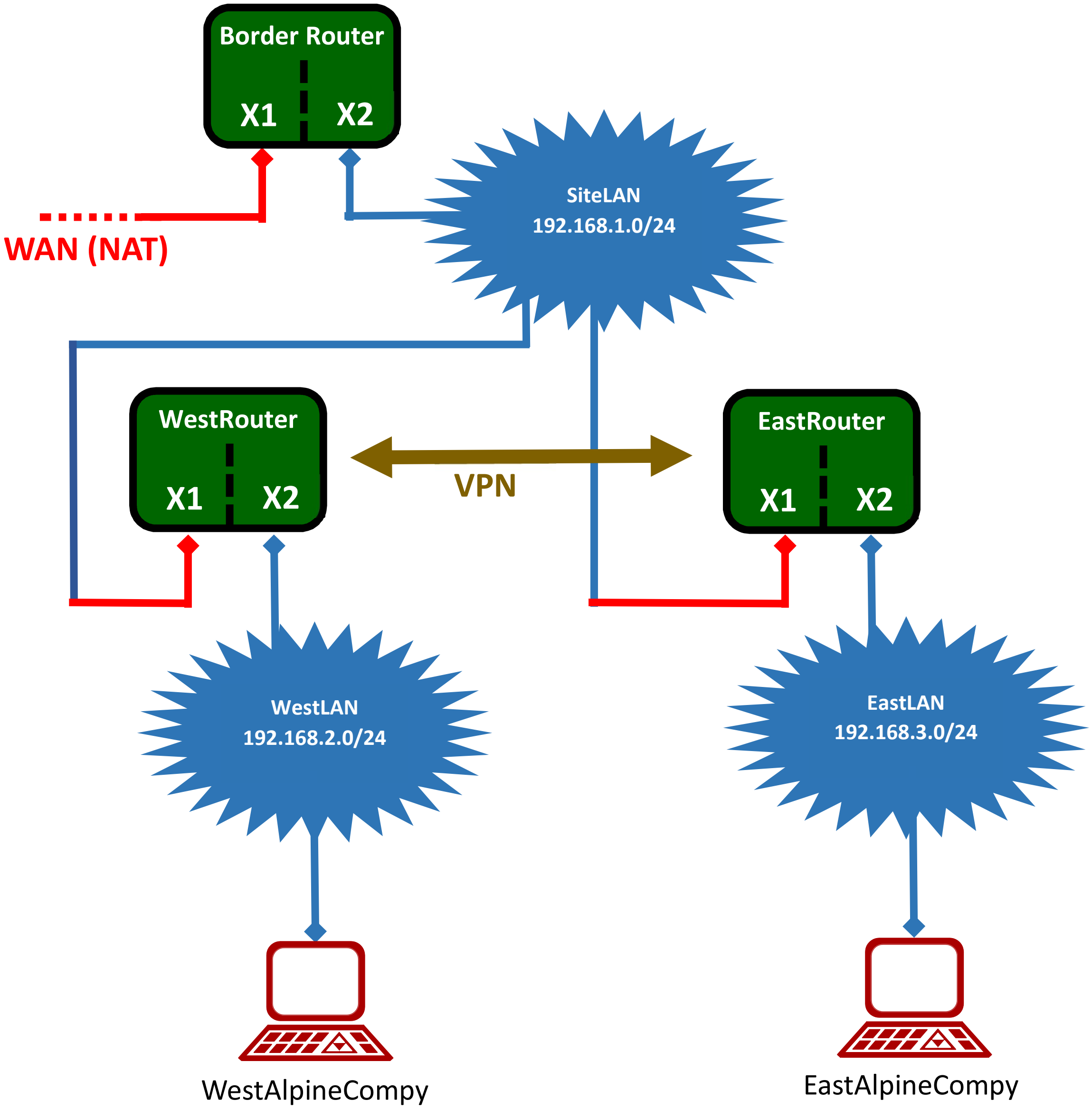
   It's possible that you receive an error message about USB, where the error message suggests that you need to install a VirtualBox extension pack. This is very easy to do! Simply go here and download the Extension pack that matches your installed VirtualBox version (Help -> About VirtualBox OR VirtualBoxVM -> About VirtualBox MV):

   https://www.virtualbox.org/wiki/Download_Old_Builds_5_2

   Once you have the file downloaded, which will have a .extpack file extension, just double click on it to install it. That should enable you to run the VMs.

   Note that I am developing these labs and VMs with VirtualBox version 5.2.6.

Here is a picture of the network that we are going to set up - feel free to come back and stare at it anytime:



2. The current state of this network is that the BorderRouter and EastRouter are fully set up, as are the WestLAN and EastLAN computers. However, the WestRouter needs to be created, and the VPN doesn't exist. Let's first get our bearings by checking the LAN/switch configurations for the Lab5 VMs
3. Verify that the Network settings for the five Lab5 VMs are set up by ensuring that the following LAN/switches are configured as follows in the VirtualBox Machine Settings -> Network controls (adapters not specified below should be disabled):
   a. Lab5_BorderRouter
      i. Adapter 1
         1. Attached to: NAT
      ii. Adapter 2
         1. Attached to: Internal Network
         2. Name: SiteLAN
   b. Lab5_WestRouter
      i. Adapter 1

1. Attached to: Internal Network
2. Name: SiteLAN

ii. Adapter 2

1. Attached to: Internal Network
2. Name: WestLAN

c. Lab5_EastRouter

i. Adapter 1

1. Attached to: Internal Network
2. Name: SiteLAN

ii. Adapter 2

1. Attached to: Internal Network
2. Name: EastLAN

d. Lab5_WestAlpineCompy

i. Adapter 1

1. Attached to: Internal Network
2. Name: WestLAN

e. Lab5_EastAlpineCompy

i. Adapter 1

1. Attached to: Internal Network
2. Name: EastLAN

4. Start up **four** of the five Lab VMs in this manner: Lab5_BorderRouter (then wait until it boots - it'll present a menu of numeric options you can choose from), EastRouter (then wait until it boots), WestAlpineCompy, and EastAlpineCompy. Finally, start up the CentOS_GUI_Reference VM. Do NOT start up the WestRouter yet.
5. Reminder: the pfSense routers will capture your mouse! You need to push the Host key (the right control button on Windows) to make it let go. This key is listed in the bottom right hand corner of the window.
6. From the WestAlpineCompy, confirm that you cannot ping www.google.com.
7. From the EastAlpineCompy, confirm that you can ping www.google.com. This works because the routers upstream are all running.
8. From the BorderRouter, type 8, then hit return. This will place you into a shell. Confirm that you can ping www.google.com. Then exit out of the shell by typing "exit".
9. Now, let's install pfSense into the WestRouter.

a. In the VB Settings -> Storage menu for WestRouter, insert the pfSense .ISO file, then start the VM.

b. Hit Accept to accept the EULA, then choose the Install option and hit OK.

c. Hit Select to "Continue with default keymap".

d. Choose the "Auto (UFS)" option for a "Guided Disk Setup". This will format the hard drive for us. This should take just a minute or two.

e. Hit No on the next screen, labeled "Manual Configuration".

f. It will next come to a screen that says the installation was complete. At this time, go ahead and eject the pfSense .ISO from the drive (if it asks, hit "Force Unmount"), and then click the Machine menu and hit Reset. The VM will now boot into pfSense

10. We can now run through the initial wizard for configuring pfSense:

a. Answer "n" to the first question about VLANs. We won't be setting those up.

b. The next question asks for the name of the WAN interface. It will be either "vtnet0" or "vtnet1". In order to figure out which is which, open up the network settings for the VM, and expand the "Advanced" arrows on adapters 1 and 2. Compare the MAC addresses listed to the table given in the VM, and then type in the name of the adapter for whichever one matches the MAC address of VB Adapter 1. This will probably be "vtnet0". Hit enter.

c. Next, type in the other adapter name for the LAN (e.g. "vtnet1", if you previously put in "vtnet0" for the WAN) and hit return.

d. Verify the setting is correct, then enter "y" and hit return.

   e. Next, you'll be placed at the main CLI menu for pfSense! This gives us a shell which is extremely useful (as you saw above), but we really can't do much configuration here: for that, we'll need a GUI. In order to make the GUI function, we need to be able to connect to it, and that means configuring the LAN adapter here in the CLI. You should see, above the menu, that the WAN interface has already received an IP address of 192.168.1.110/24 via DHCP: this is programmed in to the BorderRouter.
   f. Type 2, and hit enter. This allows us to pick an interface to configure. Type 2 again, and hit enter, as we need to set up the LAN interface.
   g. Here is the information you need for the next bunch of questions:
      i. LAN IPv4 address: 192.168.2.1
      ii. LAN IPv4 subnet bit count: 24
      iii. For the question that says "For a LAN, press <ENTER> for none", do just that. We don't need to configure a gateway here, as that's a WAN-related function.
      iv. LAN IPv6 address: Hit enter for none.
      v. Hit "y" to configure a DHCP server.
      vi. Starting IPv4 client address range: 192.168.2.10
      vii. Ending IPv4 client address range: 192.168.2.100
      viii. Hit "y" to "revert to HTTP as the webConfigurator protocol.
   h. It'll confirm the LAN interface's IP is now 192.168.2.1. This router is ready for a GUI!

11. In the CentOS GUI VM's VB settings, connect this VM to the WestRouter by changing the Internal Network to which it's attached to "WestLAN" and hitting OK.
12. Fire up a Terminal in the VM by clicking the menu up at the top and choosing Applications -> Favorites -> Terminal. Enter this line to reset the network adapter, and get a DHCP address from WestRouter:

```
$ sudo systemctl restart network.service
```

13. Now, we need to do some configuration to the router.
    a. In Firefox, connect to 192.168.2.1, and login. The default user name is "admin", and the password is "pfsense"
    b. Click Next at the first and second screen.
    c. On the third, asking for General Information, type in "8.8.8.8" and "8.8.4.4" for the Primary DNS Server and Secondary DNS Server, respectively, then click Next.
    d. Click Next on the time server screen.
    e. The next screen is for configuring the WAN interface. Scroll all the way to the bottom and uncheck the "Block RFC1918 Private Networks" checkbox. This will enable you to be able to ping WestRouter from the WAN (i.e. from BorderRouter and EastRouter), once we set up an additional firewall rule. Click Next.
    f. On the next screen for configuring the LAN interface, click Next.
    g. For the WebUI Password screen that appears next, use "password" for the password, and click Next.
    h. Finally, hit Reload, and wait a few moments.
14. In a new Firefox tab, make sure you can browse the internet!
15. OK! Now the router is mostly set up. Let's get the WestAlpineCompy VM into action. When you're logged into it, go ahead and reboot it from the command line (this is the fastest way to have it get a new IP):

```
# reboot
```

16. When you're logged back in to the WestAlpineCompy, check to see that you have an IP address by entering:

```
# ip a
```

    You should see an address in the 192.168.2.X subnet, probably 192.168.2.10.
17. From the command line, ping www.google.com to ensure you've got a full internet connection.

18. Now, we need to correct a small issue that is caused by the VM environment. We need to increase the size of a memory table so that some of the following changes we're about to make will actually work. In the CentOS GUI VM, logged into the WestRouter:
    a. Click on System -> Advanced
    b. Click on the Firewall & NAT tab (in red letters)
    c. In the "Firewall Maximum States" and "Firewall Maximum Table Entries", type in 300000.
    d. Scroll to the bottom, and hit Save. I know this doesn't mean much, but trust me: some of what's to come won't work if we don't expand that table. :)
19. Our next step is to place WestAlpineCompy onto a permanent IP address. As it stands right now, it's likely to shift around, since it's just dynamically assigned. We want to make sure that it doesn't move around on us! As you've seen, the WestRouter itself already received an assigned IP address from BorderRouter. Let's do the same for WestAlpineCompy.
    a. From within the pfSense GUI in Firefox, click the Status menu, and select DHCP Leases. This will show us all the connected devices, which should be our alpine VM and the CentOS GUI VM. You can see that the Hostname and Description fields aren't very useful.
    b. On the right side of the entry that has the same IP address as WestAlpineCompy (see step 16, above), click the white plus on the right side: this is the Add Static Mapping control.
    c. In the screen that appears, enter the following, leaving all unspecified fields alone, then clicking Save at the bottom:
        i. Client Identifier: WestAlpineCompy
        ii. IP Address: 192.168.2.101 (remember that the DHCP range goes up to 100; our address here has to be *outside* that dynamic range).
        iii. Hostname: WestAlpineCompy
        iv. Description: WestAlpineCompy
    d. Hit the green Apply Changes button
    e. Reboot the WestAlpineCompy VM, and ensure it picks up 192.168.2.101! Isn't it amazing how fast alpine boots?

    QUESTION: Go answer Question 1 now.

20. Next, we're going to enter in a Firewall rule so that we can ping the WAN interface of WestRouter from EastAlpineCompy. Fire up the EastAlpineCompy VM and login.
    a. Verify that you can't ping WestAlpineCompy from EastAlpineCompy, because they are on different subnets:

    ```
    # ping 192.168.2.101
    ```

    This also should fail:

    ```
    # ssh 192.168.2.101
    ```

    Pinging the WAN interface of WestRouter from EastAlpineCompy should also fail, because WAN interfaces by default drop/ignore/discard all incoming packets that aren't explicitly allowed:

    ```
    # ping 192.168.1.110
    ```

    b. To enable being able to ping the WAN, let's set up the Firewall rule to allow it. Back in our CentOS GUI VM, on the pfSense web page in Firefox, click on Firewall -> Rules. Note that no rules are defined, so everything that hits the WAN interface is dropped! Click either of the green Add buttons at the bottom, and enter this information (ignoring all unspecified fields):

        i. Action: Pass
        ii. Interface: WAN
        iii. Protocol: ICMP (this is what a PING packet is; it's not UDP or TCP)
        iv. Source: any
        v. Destination: WAN address
    c. Hit Save, and then Apply Changes, and verify that you can now ping the WAN interface of WestRouter from EastAlpineCompy:

```
# ping 192.168.1.110
```

21. Next, we're going to enter in some Firewall and NAT rules so that we can SSH into WestAlpineCompy from EastAlpineCompy using port-forwarding. It is not possible to port-forward a ping through the firewall, but we will port-forward the SSH service. Back in our CentOS GUI VM, on the pfSense web page in Firefox, click on Firewall -> NAT and follow this procedure to set up the port forward to WestAlpineCompy.
    a. Click either of the green Add buttons at the bottom, and enter this information (ignoring all unspecified fields):
        i. **Interface:** WAN
        ii. **Protocol:** TCP
        iii. **Source:** any
        iv. **Destination:** WAN address
        v. **Destination port range:** From port Other/Custom: 2222, and To port Other/Custom: 2222 (*We're going to use 2222 as the port that we'll trigger the router with, to make it forward to the normal SSH port on our target PC)*
        vi. **Redirect target IP**: 192.168.2.101 (*This is WestAlpineCompy)*
        vii. **Redirect target port:** Port: Other, and Custom: 22 (*This is the port that the SSH commands coming in on 2222 will be translated into, which is a good thing: the ssh server on WestAlpineCompy is only listening on port 22!)*
        viii. **Description:** Port-forward SSH to WestAlpineCompy
    b. Click Save at the bottom, and then hit Apply Changes at the top.
    c. Even though the port is now forwarded, the Firewall for this traffic has to be opened up before it'll work! Fortunately, in pfSense, this is automatically done for us. Click on the Firewall -> Rules menu, and examine the list: there's a new entry that allows this traffic through the firewall.
    d. Let's test out our SSH from EastAlpineCompy to WestAlpineCompy. Note that we have to specify that the SSH command operates on port 2222:

```
# ssh root@192.168.1.110 -p 2222
```

22. Next, we can allow SSH access to WestRouter itself like this:
    a. On the pfSense web GUI inside our CentOS GUI VM, click on the menu System -> Advanced, and select the Admin Access tab (in red letters).
    b. Scroll down to the Secure Shell section, and check the box labeled "Enable Secure Shell". Note that we're leaving the SSH port for this router at the traditional 22.
    c. Scroll down to the bottom, and hit Save.
    d. This time, the Firewall Rule was not automatically opened for us to enable this access. Click on Firewall -> Rules, and note no rule is provided for the WAN interface to allow SSH. Click either of the green Add buttons at the bottom, and enter this information (ignoring all unspecified fields):
        i. **Action:** Pass
        ii. **Interface:** WAN
        iii. **Protocol:** TCP
        iv. **Source:** any

        v. **Destination:** WAN address
        vi. **Destination port range:** From port Other/Custom: 22, and To port Other/Custom: 22.

    e. Hit Save, and then Apply Changes, and verify that you can now SSH to the WAN interface of WestRouter from EastAlpineCompy. Note that you don't specify a port, so it'll use the traditional port 22:

```
# ssh root@192.168.1.110
```

Now, this will likely fail with an error warning about a key verification failure. To fix this, remove the storage file for keys, which will then allow the connection:

```
# rm /root/.ssh/known_hosts
# ssh root@192.168.1.110
```

That'll present you with the number-choosing menu for WestRouter! You can exit out by hitting CTRL+C.

23. At this state, we can now ping all the SiteLAN router interfaces, and we can SSH into WestRouter, EastRouter, WestAlpineCompy, and EastAlpineCompy (I already set up the East side devices for you).

QUESTION: Go answer Question 2 now.

24. Our final task will be to establish a VPN link between WestRouter and EastRouter. This will make WestAlpineCompy and WestAlpineCompy be able to target each other directly! You should have noticed that you can't ping them directly from each other:

```
# ping 192.168.3.101
```

Fails from WestAlpineCompy.

```
# ping 192.168.2.101
```

Fails from EastAlpineCompy.

To get this VPN established, follow the next steps:

    a. Connect our CentOS GUI VM to the WestLAN, and then cycle it's IP:

```
$ sudo systemctl restart network.service
```

    b. In that GUI VM, connect to WestRouter in Firefox at 192.168.2.1.
    c. Click on VPN -> OpenVPN, and click on the Servers tab (in red letters). We have to pick one of the two devices to be a "server", and another to be the "client": this terminology would make more sense if we had more than two VMs to connect, as the server would be the "hub" router. In our case, though, we'll just consider WestRouter to be the server, and EastRouter to be the client.
    d. Click the green Add button.
    e. Enter the information exactly as follows, leaving all other fields untouched:
        i. **Server mode:** Peer to Peer (Shared Key)
        ii. **Protocol:** UDP on IPv4 only
        iii. **Device mode:** tun - Layer 3 Tunnel Mode
        iv. **Interface:** WAN
        v. **Local port:** 1194
        vi. **Description:** Site VPN to East Router

vii. **Shared key:** Ensure it is checked

viii. **IPv4 Tunnel Network:** 10.0.8.0/24 *(This is a throw-away network: the parties in the VPN use this subnet to communicate amongst themselves, but you never see these addresses in flight)*

ix. **IPv4 Remote network(s):** 192.168.3.0/24 *(This is the LAN network address of EastLAN)*

f. Click Save at the bottom. This establishes the server-side of the VPN. Next, we need to open up some Firewall rules to allow traffic both onto the VPN and through the VPN tunnel.

g. Click Firewall -> Rules, and click on WAN. Click the green Add button and enter the following information to allow VPN traffic to hit WestRouter:

i. **Action:** Pass

ii. **Interface:** WAN

iii. **Protocol:** UDP

iv. **Source:** “Single host or alias” and 192.168.1.111 *(This is the WAN interface of EastRouter)*

v. **Destination:** WAN address

vi. **Destination port range:** From port Other/Custom: 1194, and To port Other/Custom: 1194

h. Click Save at the bottom, and then click Apply Changes.

i. Click on the OpenVPN tab (in red letters), and then click either of the two green Add arrows to set up the Firewall rule allowing traffic down the VPN tunnel:

i. **Action:** Pass

ii. **Interface:** OpenVPN

iii. **Protocol:** Any

iv. **Source:** any

v. **Destination:** any

j. Click Save, and hit the green Apply Changes button.

k. Before we can leave WestRouter, we need to go get the VPN shared key that was automatically generated for us! Click on VPN -> OpenVPN -> Servers

l. Click the little pencil Edit button on the row of our VPN.

m. Scroll down to the Shared Key box, which will have something like this in it:

```
#
# 2048 bit OpenVPN static key
#
-----BEGIN OpenVPN Static key V1-----
bb31babc1b3289db83913b8cb...
etc.
```

I recommend you copy this into a notepad (In Centos, Applications -> Accessories -> Text Editor). Make sure you copy the whole thing, including the comments at the top and bottom!

n. Exit out of this VPN server, we just needed to copy that long shared key.

o. Now, we just need to set this up over on the client side! Connect our CentOS GUI VM to the EastLAN, and then cycle it’s IP:

```
$ sudo systemctl restart network.service
```

p. In that GUI VM, connect to EastRouter in Firefox at 192.168.3.1.

q. Click on VPN -> OpenVPN, and click on the Clients tab (in red letters). EastRouter will be the “client”.

r. Click the green Add button.

s. Enter the information exactly as follows, leaving all other fields untouched:

i. **Server mode:** Peer to Peer (Shared Key)

ii. **Protocol:** UDP on IPv4 only
iii. **Device mode:** tun - Layer 3 Tunnel Mode
iv. **Interface:** WAN
v. **Local port:** *<leave this blank>*
vi. **Server host or address:** 192.168.1.110 *(This is the WAN interface of WestRouter)*
vii. **Description:** Site VPN to West Router
viii. **Auto-generate (under Cryptographic Settings):** Uncheck, which will open up the Shared Key box.
ix. **Shared Key:** Paste in the big long key you copied from the server. Be careful not to add any newlines, etc. to it!
x. **IPv4 Tunnel Network:** 10.0.8.0/24 *(This is that throw-away network: the parties in the VPN use this subnet to communicate amongst themselves, but you never see these addresses in flight)*
xi. **IPv4 Remote network(s):** 192.168.2.0/24 *(This is the LAN network address of WestLAN)*

t. Click Save at the bottom. This establishes the server-side of the VPN. Next, we need to open up some Firewall rules to allow traffic both onto the VPN and through the VPN tunnel. This is the same as we did for the server.

u. Click Firewall -> Rules, and click on WAN. Click the green Add button and enter the following information to allow VPN traffic to hit WestRouter:
i. **Action:** Pass
ii. **Interface:** WAN
iii. **Protocol:** UDP
iv. **Source:** "Single host or alias" and 192.168.1.110 *(This is the WAN interface of WestRouter)*
v. **Destination:** WAN address
vi. **Destination port range:** From port Other/Custom: 1194, and To port Other/Custom: 1194

v. Click Save at the bottom, and then click Apply Changes.

w. Click on the OpenVPN tab (in red letters), and then click either of the two green Add arrows to set up the Firewall rule allowing traffic down the VPN tunnel:
i. **Action:** Pass
ii. **Interface:** OpenVPN
iii. **Protocol:** Any
iv. **Source:** any
v. **Destination:** any

x. Click Save, and hit the green Apply Changes button.

y. That should do it! You should now be able to ping directly from WestAlpineCompy to WestAlpineCompy and vice versa! These two computers could be thousands of miles apart, and yet logically on the network, they would appear as accessible. Amazing!

QUESTION: Go answer Question 3 now.

Note: Sometimes one of these VM routers won't give out an address via DHCP. You can see this by noticing that the client VM attached to it won't get an IP address that you've programmed it to give out. To fix this, follow the steps from 10.f through 10.h, above, then reboot the router VM. This won't reset any of the other settings, but it should fix the DHCP problem.

Questions

1) Get the TAs initials, showing that WestAlpineCompy has an IP address of 192.168.2.101. *(5 points)*
2) Which devices are targeted by these commands? Give the name of the VM of each (e.g. "WestRouter") *(2 points each)*:

```
a. # ping 192.168.1.111
b. # ping 192.168.1.110
c. # ssh root@192.168.1.111 -p 2222
d. # ssh root@192.168.1.110
e. # ssh root@192.168.1.110 -p 2222
f. # ssh root@192.168.1.111
```

3) Get the TAs initials, showing that WestAlpineCompy and EastAlpineCompy can ping each other *(23 points)*

You’re done! To receive credit for this lab, you must turn in your answer sheet. Great work!