

**Computer Science 328 -01**

## Lab Assignment 6

# Due Date: Saturday, March 9, 2024 (before midnight)

**Objectives:**

1. **Other network Linux commands:** *traceroute, arp, netstat, tcpdump*
2. **Web-based System Management – Webmin and Cockpit**
3. **Creating subnets using VM1 as a static router**

**Task 1: Other network Linux commands:**  *traceroute, arp, netstat, tcpdump*

* Please refer to <https://www.geeksforgeeks.org/traceroute-command-in-linux-with-examples/>

Open a terminal in your VM2, and install the traceroute package first:  
 *$* *sudo apt update*  
 *$ sudo apt install traceroute*

Next run the traceroute command to find out the path, four response times between hops using max number of 10 hops from your VM2 to the site [google.com](http://www.google.com). Take the screen shot of your VM2’s terminal (Note: If you see responses as a series of asterisks (\*) between hops, that’s the timeout signals due to firewall setup by blocking requests from routers, and that’s okay.). Insert your screen shot below:

|  |
| --- |
|  |

* Pleae refer to <https://www.geeksforgeeks.org/arp-command-in-linux-with-examples/>

Before you do anything between your VMs, run “*arp -a*” command on both VM2 and Kali VMs. Take a screenshot from both VMs’ terminal and insert them below:

|  |
| --- |
| Screen shot from VM2:     Screen shot from Kali VM: |

Now use “ping” command to ping between VM2 and Kali (i.e. ping from kali to VM2 and then ping from VM2 to Kali). Next run “*arp -a*” again, and take a screen shot on both VM2’s and Kali’s terminal respectively.

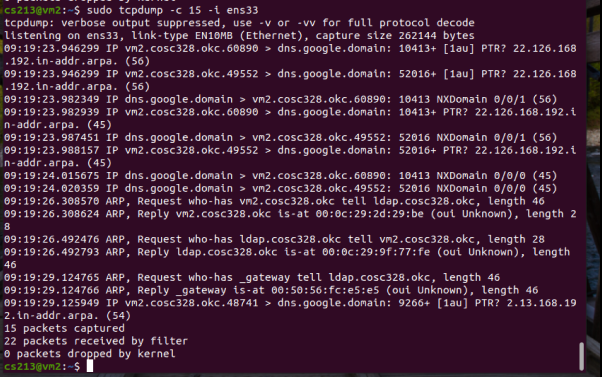
|  |
| --- |
| Screen shot from VM2:    Screen shot from Kali:    Please explain what happened when you pinged between the two VMs in terms of how ARP actually worked on this situation:  because arp -a command is used to show all the ip addresses converted to MAC address when two system are interacting with each other and when VM2 ping Kali VM both addresses are cached, therefore we could see the addresses showing up after ping |

* Please refer to <https://www.tecmint.com/20-netstat-commands-for-linux-network-management/>

Run “*netstat -a | more*” command on VM2 to find out all packet types being used for network connection between network devices within the network. Take a screen shot and insert your screen shot below:

|  |
| --- |
|  |

* Please refer to [https://www.tecmint.com/12-tcpdump-commands-a-network-sniffer-tool/#:~:text=The%20following%20command%20with%20option,ssh%20%3E%20172.16](https://www.tecmint.com/12-tcpdump-commands-a-network-sniffer-tool/" \l ":~:text=The%20following%20command%20with%20option,ssh%20%3E%20172.16).

  
  
Now run the *tcpdump* command on your VM2 to capture **15** packets on interface **ens33.** Take a screen shot and insert the screen shot below:

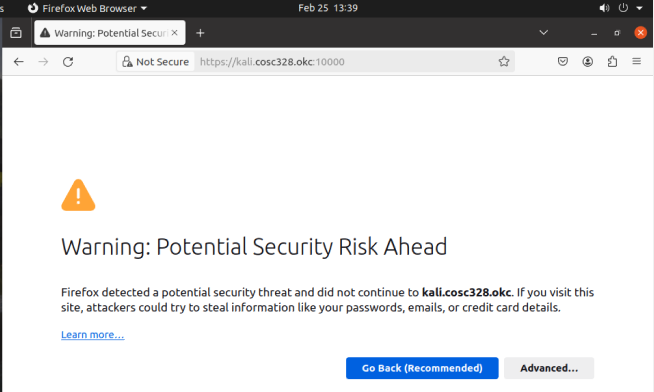
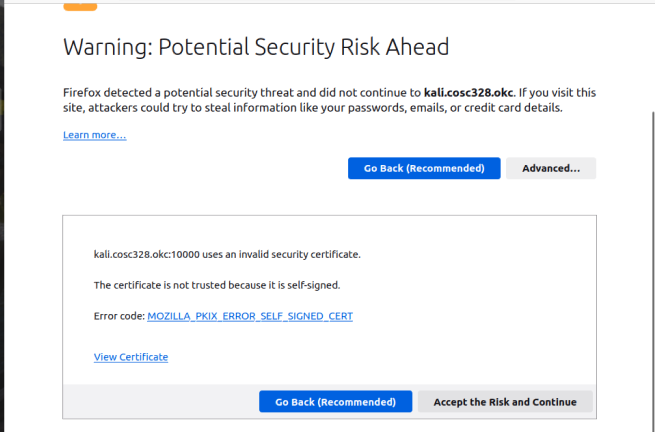
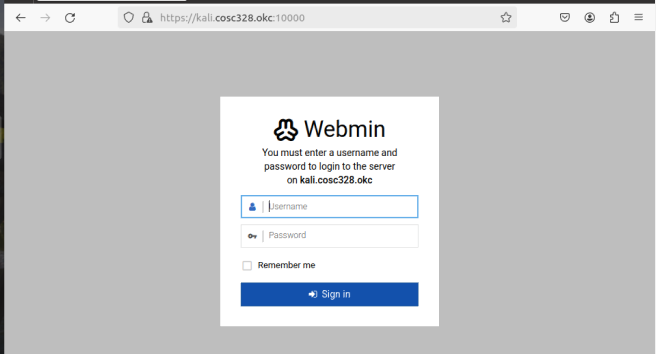
|  |
| --- |
|  |

  
\

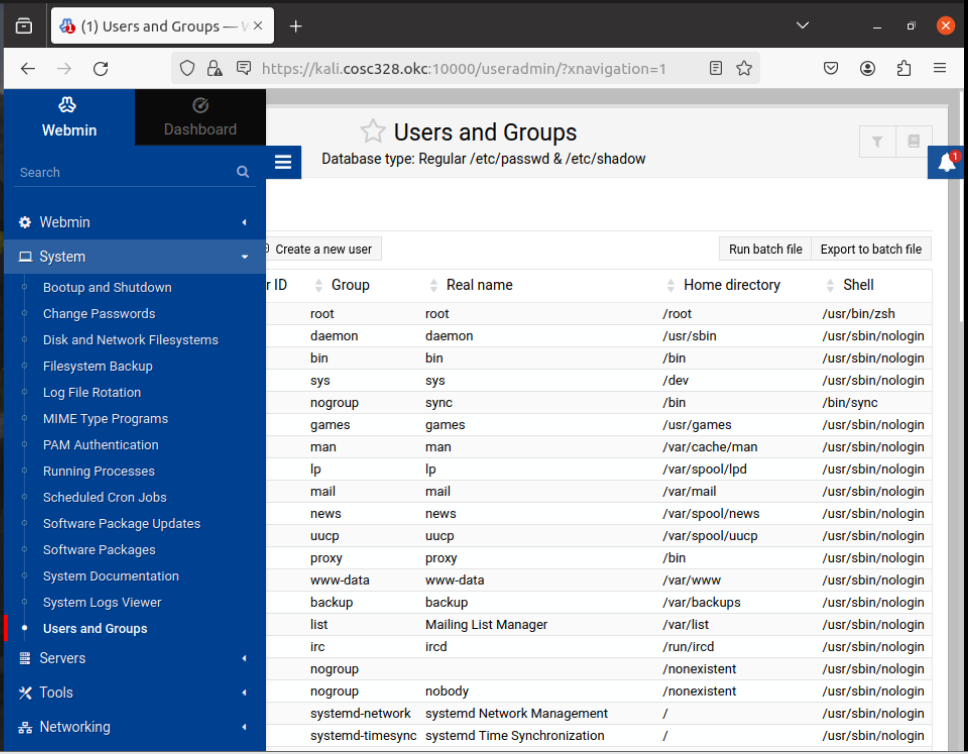
Now run the tcpdump command again on your VM2 to capture **2** packets on interface **ens33** and display the captured packets in HEX and ASCII formats**.** Take a screen shot and insert the screen shot below:

|  |
| --- |
|  |

**Task 2: Web-based System Management – Webmin and Cockpit**

* Open a terminal in Kali VM. Follow the same steps from the following web page to install and configure Webmin on Kali VM:  
    
  <https://phoenixnap.com/kb/install-webmin-on-ubuntu>
* Don’t try to change root’s password. Use the username “Kali” instead of root, and the same password “kali” to log into webmin.  
    
  - Once Webmin has been setup on Kali VM, **open a web browser in VM2** and navigate to <https://kali.cosc328.okc:10000> (or using Kali’s IP), and you’ll see a screen like this for the first connection:  
    
    
    
    
    
    
  Click on **Advanced..** button =>  
    
    
    
  Click on “Accept the risk and continue” button (Once you accepted the self-signed SSL/TLS certificate, you won’t see the same warning again for all subsequent connections)=>  
    
  

Login as **kali** user (with password: **kali**).

****

* You can do a whole lots of thing as system administer with Webmin. Now you only need to create a new group and a new user.

|  |
| --- |
| Take a screen shot to show a new group (WebminGroup) has been created in Webmin:    Take a screen shot to show a new user (webminuser which belongs to WebminGroup as a Secondary group) has been created in Webmin:    Take a screen shot to show that the new user (webminuser) logged into Kali successfully by showing the user’s home directory with the pwd command: |

- Now open a terminal in Ubuntu VM2. Install and enable **Cockpit** as follows:  
  
$ sudo apt install cockpit

$ sudo apt install firewalld

Next, the Cockpit socket service needs to be enabled:

$ sudo systemctl enable --now cockpit.socket

Finally, the necessary ports need to be opened on the firewall to allow remote browser connections to reach Cockpit if a firewall is enabled on your system. If ufw is enabled:

$ sudo ufw allow 9090

If firewalld is enabled:

$ sudo firewall-cmd --add-service=cockpit --permanent

## $ sudo firewall-cmd --reload

## User Management with Cockpit

Once Cockpit has been setup on VM2, **open a web browser in Kali VM** and navigate to <https://vm2.cosc328.okc:9090> (or using VM2’s IP) to access the Cockpit sign-in screen. Check this site for more details:

<https://www.answertopia.com/ubuntu/an-overview-of-the-ubuntu-cockpit-web-interface/>

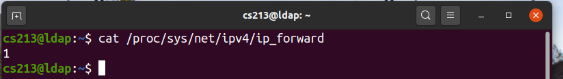
* Create a new group and a new user with Cockpit.

|  |
| --- |
| Take a screen shot to show a new group (CockpitGroup) has been created in Cockpit:    Take a screen shot to show a new user (cockpituser which belongs to CockpitGroup as a Secondary group) has been created in Cockpit:    Take a screen shot to show that the new user (cockpituser) logged into VM2 successfully by showing the user’s home directory with the pwd command: |

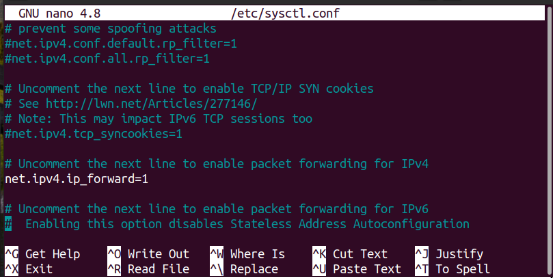
**Task 3: Creating subnets using VM1 as a static router**

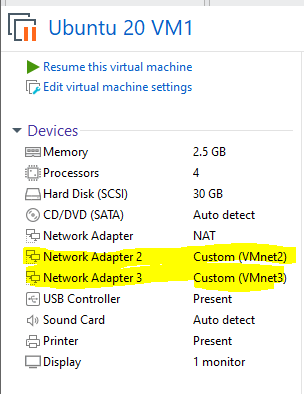
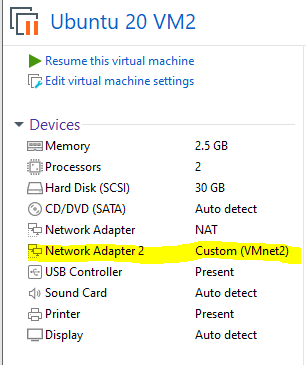
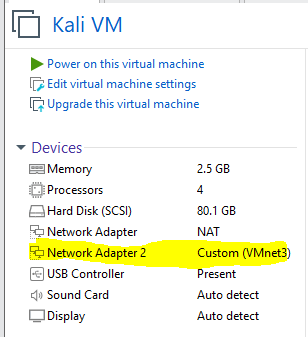
Please refer to the concept and information introduced in “***How to turn a Linux machine into a static router.pdf***”. Without using physical hardware devices, we can use virtual machines (e.g. Ubuntu VM1, VM2 and Kali VM) to achieve the same task with the features available from VMWare Workstation virtualization software.

* Make sure that “ipv4\_forward” feature is already turned on (by Network Manager) by issuing a command like this:  
   $ cat /proc/sys/net/ipv4/ip\_forward  
   1  
  If the returned value is 1, that means “Ipv4 Forwarding” feature is already turned on!

  
  
If the returned value is 0, then we need to change the value from 0 to 1 either temporary or permanently.

To change “ipv4 forward” feature temporary, this is how we do it manually every time we boot up the machine at a terminal:  
$ echo 1 > /proc/sys/net/ipv4/ip\_forward  
  
To change “ipv4 forward” feature permanently, we need edit /etc/sysctl.conf and uncomment (i.e. remove the # sign) the line *“#net.ipv4.ip\_forward=1”* from the conf file.

  
  
Save the conf file, and every time we power on our VM1 to restart, the “ipv4 forward” feature will be turned on automatically.

* Add two additional Network Adapters into Ubuntu VM1   
    
  
* Add one additional Network Adapter into Ubuntu VM2  
    
  
* Add one additional Network Adapter into Kali VM  
    
  

Now please refer to ***“Screenshots of netplan settings for all VMs.pdf”*** to complete your virtual subnets setup with VMWare Workstation.

**If you use a different virtualization software**, you should try the similar way to set up your virtual subnets and see whether you can get it done successfully. Here is a link showing how to create a virtual network on VMWare Fusion:

<https://graspingtech.com/vmware-fusion-add-networks/>

I’ll award 10% bonus for a successful attempt on this part if you can complete the task successfully with a different virtualization software other than VMWare Workstation.

|  |
| --- |
| **Get your subnets setup done successfully?**  If **yes**, take screen shot(s) to show:   1. All IP addresses assigned on the two network adapters in Ubuntu VM2 2. All IP addresses assigned on the two network adapters in Kali VM 3. Successful ping from kali to Ubuntu VM2 using the **second** NIC IP’s addresses 4. Successful ping from Ubuntu VM2 using to Kali using the **second** NIC IP’s addresses     **If your answer is “No**”, tell me what you have done and how far you have accomplished. Some screen shots would be good. |

**Submitting your work:**

Export this Word document with all your answers and screen shots as PDF format, and then submit your PDF file via Lab 6 assignment tab on Moodle by *Saturday, March 9, 2024 (midnight).*