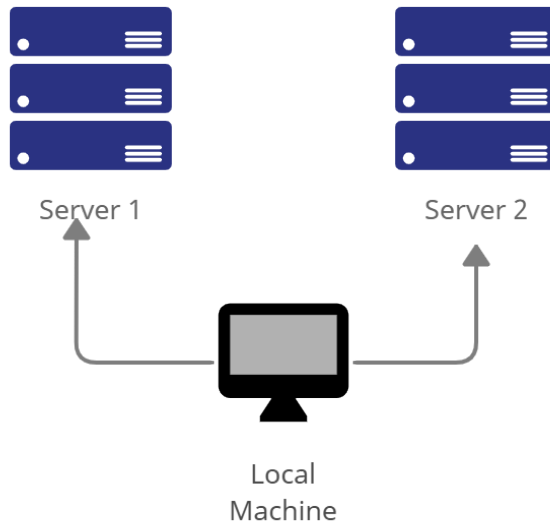


Name: Yuri P. Nollan	Date Performed: 08/17/2023
Course/Section: CPE31S6	Date Submitted: 08/17/2023
Instructor: Dr. Jonathan Taylar	Semester and SY: 1st sem 2023-2024
Activity 1: Configure Network using Virtual Machines	
1. Objectives: 1.1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox 1.2. Set-up a Virtual Network and Test Connectivity of VMs	
2. Discussion: Network Topology: Assume that you have created the following network topology in Virtual Machines, <i>provide screenshots for each task</i> . (Note: <i>it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine</i>).	
 <pre> graph TD LocalMachine[Local Machine] --> Server1[Server 1] LocalMachine --> Server2[Server 2] </pre> <p>The diagram illustrates a network topology. At the bottom center is a computer icon labeled "Local Machine". Two lines extend upwards from the "Local Machine" to two server stacks. The left server stack is labeled "Server 1" and the right server stack is labeled "Server 2". Each server stack consists of three blue rectangular blocks, each with a white dot and three horizontal lines, representing a server rack.</p>	
Task 1: Do the following on Server 1, Server 2, and Local Machine. In editing the file using nano command, press control + O to write out (save the file). Press enter when asked for the name of the file. Press control + X to end.	
1. Change the hostname using the command <i>sudo nano /etc/hostname</i> 1.1 Use server1 for Server 1	

```
GNU nano 2.9.3 /etc/hostname
server1
```

1.2 Use server2 for Server 2

```
GNU nano 2.9.3 /etc/hostname
server2
```

1.3 Use workstation for the Local Machine

```
GNU nano 2.9.3 /etc/hostname
workstation
```

2. Edit the hosts using the command *sudo nano /etc/hosts*. Edit the second line.

2.1 Type 127.0.0.1 server 1 for Server 1

```
GNU nano 2.9.3 /etc/hosts
127.0.1.1 server1
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

2.2 Type 127.0.0.1 server 2 for Server 2

```
127.0.1.1      server2
```

```
# The following lines are desirable for IPv6 capable hosts
```

```
:::1          ip6-localhost ip6-loopback
```

```
fe00::0 ip6-localnet
```

```
ff00::0 ip6-mcastprefix
```

```
ff02::1 ip6-allnodes
```

```
ff02::2 ip6-allrouters
```

2.3 Type 127.0.0.1 workstation for the Local Machine

```
GNU nano 2.9.3
```

```
/etc/hosts
```

```
Modified
```

```
127.0.1.1      workspace
```

```
# The following lines are desirable for IPv6 capable hosts
```

```
:::1          ip6-localhost ip6-loopback
```

```
fe00::0 ip6-localnet
```

```
ff00::0 ip6-mcastprefix
```

```
ff02::1 ip6-allnodes
```

```
ff02::2 ip6-allrouters
```

Task 2: Configure SSH on Server 1, Server 2, and Local Machine. Do the following:

1. Upgrade the packages by issuing the command *sudo apt update* and *sudo apt upgrade* respectively.
2. Install the SSH server using the command *sudo apt install openssh-server*.

```
workspace@workstation-VirtualBox:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  liblvm7
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere rssh ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 0 not upgraded.
Need to get 637 kB of archives.
After this operation, 5,320 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 ncurses-ter
m all 6.1-1ubuntu1.18.04.1 [248 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openssh-sft
p-server amd64 1:7.6p1-4ubuntu0.7 [45.5 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openssh-ser
ver amd64 1:7.6p1-4ubuntu0.7 [332 kB]
Get:4 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 ssh-import-
id all 5.7-0ubuntu1.1 [10.9 kB]
Fetched 637 kB in 6s (111 kB/s)
Preconfiguring packages ...
Selecting previously unselected package ncurses-term.
(Reading database ... 162327 files and directories currently installed.)
```

3. Verify if the SSH service has started by issuing the following commands:

3.1 sudo service ssh start

3.2 sudo systemctl status ssh

```
workspace@workstation-VirtualBox:~$ sudo service ssh start
workspace@workstation-VirtualBox:~$ sudo ufw enable
Firewall is active and enabled on system startup
workspace@workstation-VirtualBox:~$ sudo ufw status
Status: active
workspace@workstation-VirtualBox:~$
```

4. Configure the firewall to all port 22 by issuing the following commands:

4.1 sudo ufw allow ssh

4.2 sudo ufw enable

4.3 sudo ufw status

```
workspace@workstation-VirtualBox:~$ sudo service ssh start
workspace@workstation-VirtualBox:~$ sudo ufw enable
Firewall is active and enabled on system startup
workspace@workstation-VirtualBox:~$ sudo ufw status
Status: active
workspace@workstation-VirtualBox:~$
```

Task 3: Verify network settings on Server 1, Server 2, and Local Machine. On each device, do the following:

1. Record the ip address of Server 1, Server 2, and Local Machine. Issue the command *ifconfig* and check network settings. Note that the ip addresses of all the machines are in this network 192.168.56.XX.

1.1 Server 1 IP address: 192.168.56. 104

1.2 Server 2 IP address: 192.168.56.105

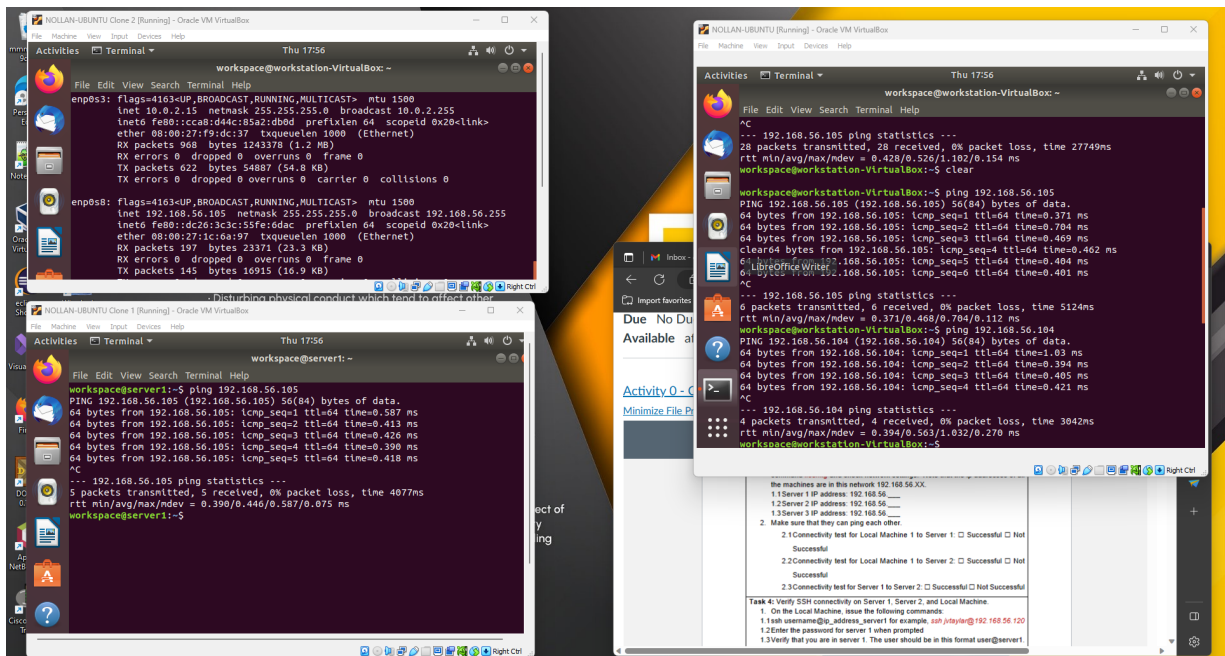
1.3 Server 3 IP address: 192.168.56.101

1.4 Make sure that they can ping each other.

2.1 Connectivity test for Local Machine 1 to Server 1: ☐ Successful

Connectivity test for Local Machine 1 to Server 2: ☐ Successful

Connectivity test for Server 1 to Server 2: ☐ Successful



Task 4: Verify SSH connectivity on Server 1, Server 2, and Local Machine.

1. On the Local Machine, issue the following commands:

1.1 *ssh* username@ip_address_server1 for example, *ssh jvtaylor@192.168.56.120*

1.2 Enter the password for server 1 when prompted

1.3 Verify that you are in server 1. The user should be in this format user@server1.

For example, *jvtaylor@server1*

```
workspace@workstation-VirtualBox:~$ ssh workspace@192.168.56.104
The authenticity of host '192.168.56.104 (192.168.56.104)' can't be established
.
ECDSA key fingerprint is SHA256:0CZnjMrFhSwWrW3QWMq8Go+0Muo7TIivmt2B8DbpdKk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.104' (ECDSA) to the list of known hosts.
workspace@192.168.56.104's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Your Hardware Enablement Stack (HWE) is supported until April 2023.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

2. Logout of Server 1 by issuing the command *control + D*.

```
workspace@server1:~$ logout
Connection to 192.168.56.104 closed.
workspace@workstation-VirtualBox:~$
```

3. Do the same for Server 2.

```
workspace@workstation:~$ ssh workspace@192.168.56.105
workspace@192.168.56.105's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Thu Aug 17 18:02:42 2023 from 192.168.56.101
workspace@server2:~$ logout
Connection to 192.168.56.105 closed.
workspace@workstation:~$
```

4. Edit the hosts of the Local Machine by issuing the command *sudo nano /etc/hosts*. Below all texts type the following:

4.1 *IP_address server 1* (provide the ip address of server 1 followed by the hostname)

4.2 *IP_address server 2* (provide the ip address of server 2 followed by the hostname)

4.3 Save the file and exit.

GNU nano 2.9.3 */etc/hosts*

```
127.0.1.1      workspace
192.168.56.104 Server 1
192.168.56.105 Server 2
# The following lines are desirable for IPv6 capable hosts
::1          ip6-localhost ip6-loopback
fe00::0      ip6-localnet
ff00::0      ip6-mcastprefix
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
```

5. On the local machine, verify that you can do the SSH command but this time, use the hostname instead of typing the IP address of the servers. For example, try to do `ssh jvtaylor@server1`. Enter the password when prompted. Verify that you have entered Server 1. Do the same for Server 2.

Reflections:

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
 - we can use the hostname in SSH commands because of the DNS. DNS is a system - based that can translate a readable domain names like the www.facebook.com into ip addresses such as 192.168.1.1 to identify each other on a network.
2. How secured is SSH?
 - SSH is secured if there are proper implementations, configurations and maintenance. It can be highly secured if the implementations of the commands has strong encryption and cannot be penetrated. Though, SSH can be penetrated if it is not properly configured or weak authentications are used.