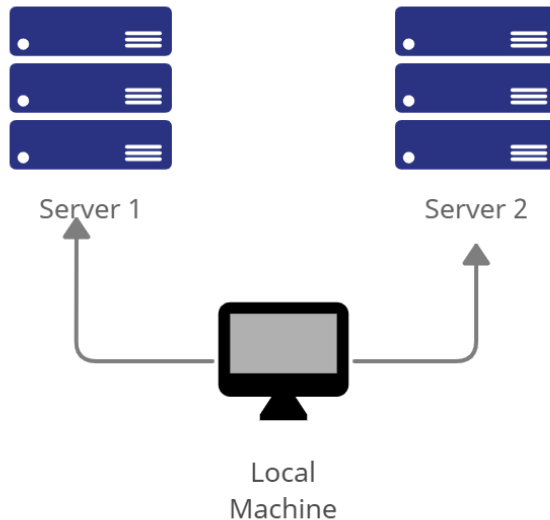


Name: Cuyugan, Ken Lester C	Date Performed: August 30,2022
Course/Section: CPE343-CPE31S22	Date Submitted: August 30,2022
Instructor: ENGR. JONATHAN TAYLAR	Semester and SY: 1ST SEM SY 2022
Activity 1: Configure Network using Virtual Machines	
1. Objectives: 1.1. Create and configure Virtual Machines 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, each ending in an arrow pointing to a stack of three server icons. The left stack is labeled "Server 1" and the right stack is labeled "Server 2". Each server icon is a blue rectangle with a white dot and three horizontal lines on the right side.</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	

```
kencu@kencu-VirtualBox:~$ sudo nano /etc/hostname
[sudo] password for kencu:
kencu@kencu-VirtualBox:~$ hostnamectl
  Static hostname: kencu-server1
  Transient hostname: kencu-VirtualBox
    Icon name: computer-vm
    Chassis: vm
  Machine ID: e3420e9689ba4a84b1031e05970fb4d6
  Boot ID: 3b4059d1d77b40a7ba67b8a2fa48f46f
  Virtualization: oracle
  Operating System: Ubuntu 22.04 LTS
    Kernel: Linux 5.15.0-46-generic
  Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
kencu@kencu-VirtualBox:~$
```

1.2 Use server2 for Server 2

```
kencu@kencu-VirtualBox:~$ sudo nano /etc/hostname
[sudo] password for kencu:
kencu@kencu-VirtualBox:~$ tlchostname
tlchostname: command not found
kencu@kencu-VirtualBox:~$ hostnamectl
  Static hostname: kencu-server2
  Transient hostname: kencu-VirtualBox
    Icon name: computer-vm
    Chassis: vm
  Machine ID: e3420e9689ba4a84b1031e05970fb4d6
  Boot ID: 826873956ac44d47897e31c4befa576a
  Virtualization: oracle
  Operating System: Ubuntu 22.04 LTS
    Kernel: Linux 5.15.0-46-generic
  Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
kencu@kencu-VirtualBox:~$
```

1.3 Use workstation for the Local Machine

```
kencu@kencu-VirtualBox:~$ sudo nano /etc/hostname
[sudo] password for kencu:
kencu@kencu-VirtualBox:~$ sudo nano /etc/hostname
kencu@kencu-VirtualBox:~$ hostnamectl
  Static hostname: kencu-workstation
  Transient hostname: kencu-VirtualBox
    Icon name: computer-vm
    Chassis: vm
  Machine ID: e3420e9689ba4a84b1031e05970fb4d6
  Boot ID: 1579ae75a6e84cd6bd4d3be1e5ec4b61
  Virtualization: oracle
  Operating System: Ubuntu 22.04 LTS
    Kernel: Linux 5.15.0-46-generic
  Architecture: x86-64
  Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
kencu@kencu-VirtualBox:~$
```

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 6.2 /etc/hosts
127.0.0.1    server 1
127.0.1.1    kencu-VirtualBox
```

2.2 Type 127.0.0.1 server 2 for Server 2

```
GNU nano 6.2 /etc/hosts
127.0.0.1    server 2
127.0.1.1    kencu-VirtualBox
```

2.3 Type 127.0.0.1 workstation for the Local Machine

```
GNU nano 6.2 /etc/hosts
127.0.0.1    workstation
127.0.1.1    kencu-VirtualBox
```

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.

Server 1

```
kencu@kencu-VirtualBox:~$ sudo apt update
Hit:1 http://ph.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://ph.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 110 kB in 1s (77.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
271 packages can be upgraded. Run 'apt list --upgradable' to see them.
kencu@kencu-VirtualBox:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
accountsservice apparmor apport apport-gtk apt apt-utils base-files
bind9-dnswriter bind9-host bind9-libs brltty cups cups-bsd cups-client
cups-common cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc
cups-server-common deja-dup dirmngr distro-info-data dnsmasq-base dpkg
evince evince-common evolution-data-server evolution-data-server-common
firmware-sof-signed fonts-opensymbol gdm3 glib2.0-accountsservice-1.0
glib2.0-gdm-1.0 glib2.0-gnome-desktop-3.0 glib2.0-gstreamer-1.0 glib2.0-gtk-3.0
glib2.0-gtk-4.0 glib2.0-harfbuzz-0.0 glib2.0-javascriptcoregtk-4.0
glib2.0-mutter-10 glib2.0-nm-1.0 glib2.0-notify-0.7 glib2.0-webkit2-4.0 gjs
gnome-control-center gnome-control-center-data gnome-control-center-faces
gnome-desktop3-data gnome-keyring gnome-keyring-pkcs11 gnome-remote-desktop
gnome-shell-common gnome-shell-extension-desktop-icons-ng
```

Server 2

```
kencu@kencu-VirtualBox:~$ sudo apt update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://ph.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://ph.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 110 kB in 17s (6,368 B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
271 packages can be upgraded. Run 'apt list --upgradable' to see them.
kencu@kencu-VirtualBox:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
accountsservice apparmor apport apport-gtk apt apt-utils base-files
bind9-dnswriter bind9-host bind9-libs brltty cups cups-bsd cups-client
cups-common cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc
cups-server-common deja-dup dirmngr distro-info-data dnsmasq-base dpkg
evince evince-common evolution-data-server evolution-data-server-common
firmware-sof-signed fonts-opensymbol gdm3 glib2.0-accountsservice-1.0
glib2.0-gdm-1.0 glib2.0-gnome-desktop-3.0 glib2.0-gstreamer-1.0 glib2.0-gtk-3.0
glib2.0-gtk-4.0 glib2.0-harfbuzz-0.0 glib2.0-javascriptcoregtk-4.0
glib2.0-mutter-10 glib2.0-nm-1.0 glib2.0-notify-0.7 glib2.0-webkit2-4.0 gjs
gnome-control-center gnome-control-center-data gnome-control-center-faces
gnome-desktop3-data gnome-keyring gnome-keyring-pkcs11 gnome-remote-desktop
gnome-shell-common gnome-shell-extension-desktop-icons-ng
```

workstation

```
kencu@kencu-VirtualBox:~$ sudo apt update
Hit:1 http://ph.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://ph.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 110 kB in 4s (27.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
255 packages can be upgraded. Run 'apt list --upgradable' to see them.
kencu@kencu-VirtualBox:~$ sudo apt upgrade
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
eld by process 3042 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
eld by process 3042 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
eld by process 3042 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
eld by process 3042 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
eld by process 3042 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
Waiting for cache lock: Could not get lock /var/lib/dpkg/lock-frontend. It is h
eld by process 3042 (unattended-upgr)
```

2. Install the SSH server using the command *sudo apt install openssh-server*.
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*
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*

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.____
 - 1.2 Server 2 IP address: 192.168.56.____
 - 1.3 Server 3 IP address: 192.168.56.____
2. Make sure that they can ping each other.
 - 2.1 Connectivity test for Local Machine 1 to Server 1: ☐ Successful ☐ Not Successful
 - 2.2 Connectivity test for Local Machine 1 to Server 2: ☐ Successful ☐ Not Successful
 - 2.3 Connectivity test for Server 1 to Server 2: ☐ Successful ☐ Not 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*
2. Logout of Server 1 by issuing the command *control + D*.
3. Do the same for Server 2.
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.
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?
 - **YOU WILL EDIT THE SSH FILE AND CHANGE THE COMMANDS**
2. How secured is SSH?
 - **To protect communication between the parties involved, the SSH protocol employs industry standard strong encryption methods such as AES.**