| Hands-on Activity 0 - Creating Virtual Machines | |
| --- | --- |
| **Name:** Torrecampo, Juan Piolo S. | **Date Performed:** August 16, 2022 |
| **Course/Section:** CPE 232 / CPE31S22 | **Date Submitted:** August 18, 2022 |
| **Instructor:** Dr. Jonathan Taylar | **Semester and SY:** 1st Sem, 2022 - 2023 |
| 1. **Objectives:**    1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox    2. Set-up a Virtual Network and Test Connectivity of VMs | |
| 1. **Discussion:**   **Network Topology:**  Assume that you have created the following network topology in Virtual Machines, *provide screenshots for each task*. (Note: *it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine*). | |
| **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 *sudo nano /etc/hostname*   2. Use server1 for Server 1   3. Use server2 for Server 2   4. Use workstation for the Local Machine      1. Edit the hosts using the command *sudo nano /etc/hosts.* Edit the second line.    1. Type 127.0.0.1 server 1 for Server 1      * 1. Type 127.0.0.1 server 2 for Server 2      * 1. Type 127.0.0.1 workstation for the Local Machine     **After Rebooting** | |
| **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.      * 1. Install the SSH server using the command *sudo apt install openssh-server*.   Server 1    Server 2    Local Machine     * 1. Verify if the SSH service has started by issuing the following commands:   2. *sudo service ssh start*   3. *sudo systemctl status ssh*   Server 1    Server 2    Local Machine     * 1. Configure the firewall to all port 22 by issuing the following commands:   2. *sudo ufw allow ssh*   3. *sudo ufw enable*   4. *sudo ufw status*   Server 1    Server 2    Local Machine | |
| **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.   2. Server 1 IP address: 192.168.56.103   3. Server 2 IP address: 192.168.56.101   4. Server 3 IP address: 192.168.56.102 (local machine)   Server 1    Server 2    Local Machine     * 1. Make sure that they can ping each other.   2. Connectivity test for Local Machine 1 to Server 1: **✓ Successful** ☐ Not Successful      * 1. Connectivity test for Local Machine 1 to Server 2: **✓ Successful** ☐ Not Successful      * 1. 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. ssh username@ip\_address\_server1 for example, *ssh* [*jvtaylar@192.168.56.120*](mailto:jvtaylar@192.168.56.120)    2. Enter the password for server 1 when prompted    3. Verify that you are in server 1. The user should be in this format user@server1. For example, *jvtaylar@server1* 2. Logout of Server 1 by issuing the command *control + D.*      1. Do the same for Server 2.      1. Edit the hosts of the Local Machine by issuing the command *sudo nano /etc/hosts.* Below all texts type the following:    1. IP\_address server 1 (provide the ip address of server 1 followed by the hostname)    2. IP\_address server 2 (provide the ip address of server 2 followed by the hostname)      * 1. Save the file and exit.  1. 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 jvtaylar@server1*. Enter the password when prompted. Verify that you have entered Server 1. Do the same for Server 2.   Local Machine to Server 1    Local Machine to Server 2 | |
| **Reflections:**  Answer the following:   * 1. How are we able to use the hostname instead of IP address in SSH commands? * We are able to achieve the use of using the IP address in SSH commands by performing the 4 in task 4 which we are editing the /etc/hosts file where we add the IP address of server1 and server2 corresponding to its name. In this way we are stating an alias to an IP address where we can use it in accessing the SSH servers using only the username of the server.   1. How secured is SSH? * SSH uses the TCP protocol in port 22 in connecting to the remote machine but this connection is not secured by default. The SSH does is it first encrypts the packets that are being sent in the receiver machine then decrypts it after accepting the packet and vice versa. The other reason why SSH is connected is because it uses a multiplex multiple connect which creates a tunnel over the TCP connection. It also uses authentication when connecting to a machine that provides a high level of security but it can be brute forced sometimes. | |
| **Honor Pledge:**  *“I affirm that I shall not give or receive any unauthorized help on this hands-on activity and that all work shall be my own. “* | |