**Systems Integration Assignment: Configuring a Server**

**Jake Bolger - C18395341**

**How-to Document Includes:** Ubuntu Server setup, commands used (explaining what they do), and where each configuration file goes on the system. Reader is assumed to know what each software package is used to do.

**Part 1:** Setting up Ubuntu on your machine.

First step is setting up Ubuntu on your VM. First download Ubuntu server. Run Virtual box and create a new VM.

Your VM settings should be as follows: type set to Linux, and version “Ubuntu (64 bit). The memory should be around 1024MB, and the hard disk should e around 10GB.

Enable your network adaptor and set to a bridged adapter. Next run your VM.

**DNS Server**

**Part 2:** Setting up a **DNS Server** for the domain example.lan with both forward and reverse lookup.

In this part you will be setting up a DNS server for the domain example.lan that allows you to do a forward and reverse lookup.

In your terminal type in the command **‘sudo apt install bind9’.** This command will install BIND for DNS tools.

Once you have run this command the daemon should start running, if not then you should run the command **‘sudo systemctl start named’** .

Next you need to create a new zone. This means you will have to change the configuration so that we can set our own IP address.

The ip address that we will set up will be **192.168.1.50** for the domain **example.lan.**

When using BIND, the config and zone files are stored in the directory /etc/bind. You now need to create a zone file for the forward lookup using the command**:**

**‘sudo nano /etc/bind/forward.example.lan’**

In this file you need to enter the following shown in the screenshot below:

Text

Description automatically generated

To save this file use the command **‘ctrl + o’** and then **‘ctrl + x’** to exit. Make sure the syntax is correct and you don’t use tab to indent as it does not recognize tab.

Next to create the revers lookup zone file use the command **‘sudo nano /etc/bind/reverse.example.lan’**

Enter the following:

Text

Description automatically generated

Save and exit as before.

The next step requires you to alter a config file. This alerts it to our new zone file. Enter the command **‘sudo nano /etc/bind/named.conf’** and enter the following into it as shown in the screenshot below:

Text

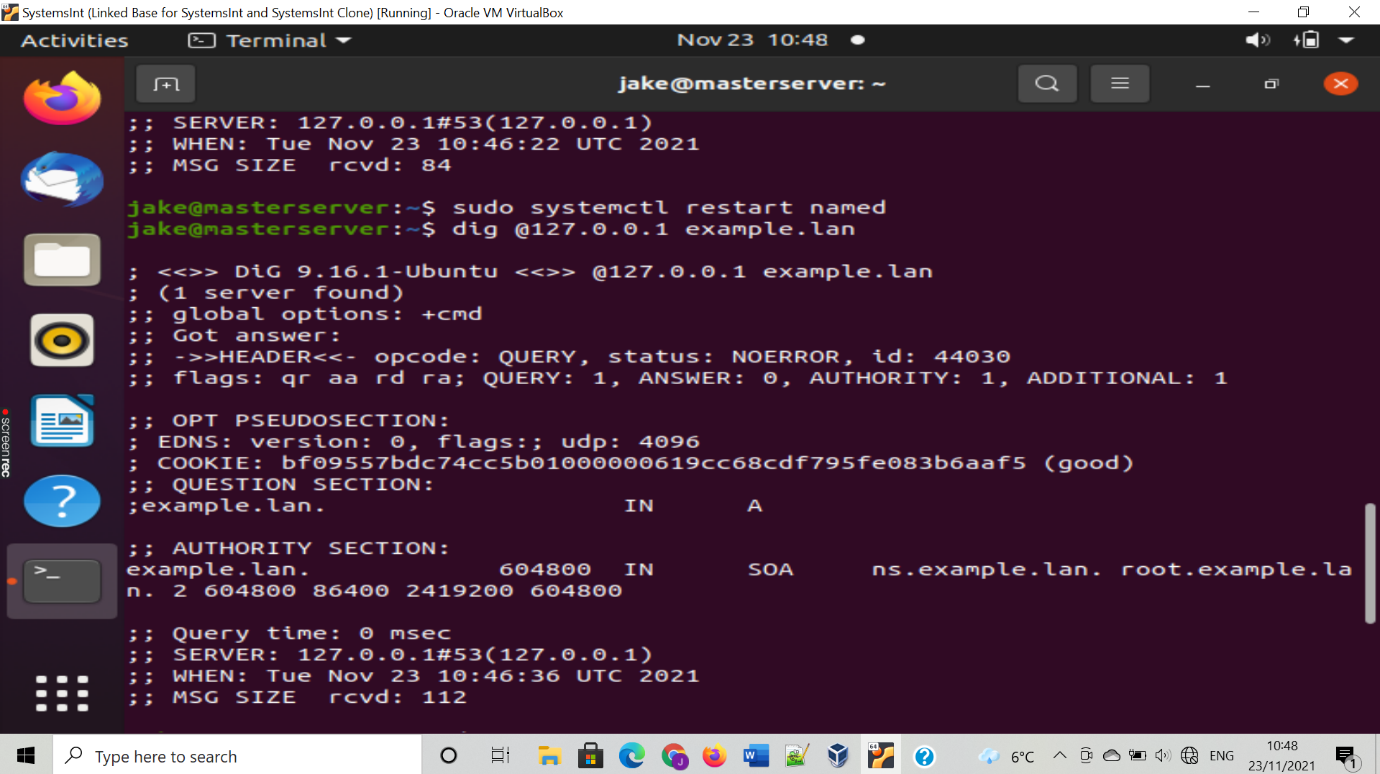
Description automatically generated

One is for forward lookup and one is for reverse. Next restart the named because we want it to read the new files. Run the command: ‘sudo systemctl restart named’.

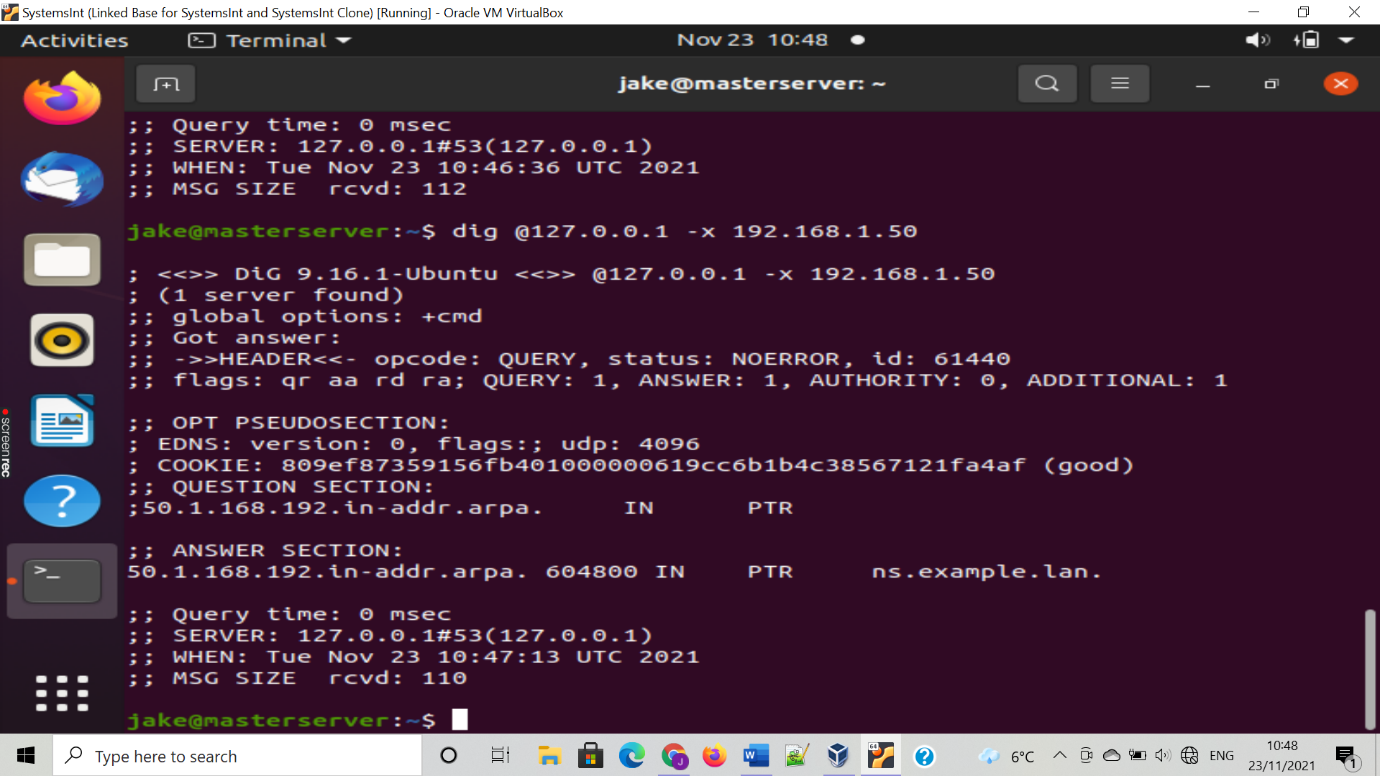
If everything worked you will be able to perform reverse and forward look up by dig your own resolver. For forward lookup use the command **‘dig @127.0.0.1 example.lan’ and for reverse use ‘dig @127.0.0.1 -x 192.168.1.50’.**

The screen shots below show what the output should be:

**DNS Server forward lookup:**



**Reverse lookup:**



**DHCP Server**

**Part 3:** in this next part you will be shown how to set up a **DHCP** server and lease clients in the IP address range **192.168.1.150** to **192.168.1.200.**

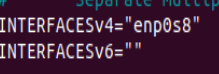
When using a DHCP server you will need to change a few settings on your VM. First enable adaptor 2 in your network, enable the adaptor and set to internal network. Save your settings and create a clone of your VM. Make sure your MAC address policy is set to **‘generate new MAC address for all network adapters’.**

Next up you want to install and configure your DHCP server. In your original VM install DHCP by using the command **‘sudo apt install isc-dhcp-server’.**

The to see the list of network interface run **‘ip link’.** This will show you interfaces that arnt available when you run **‘ifconfig’.** Write down the name of the interface that wasn’t on the ifconfig.

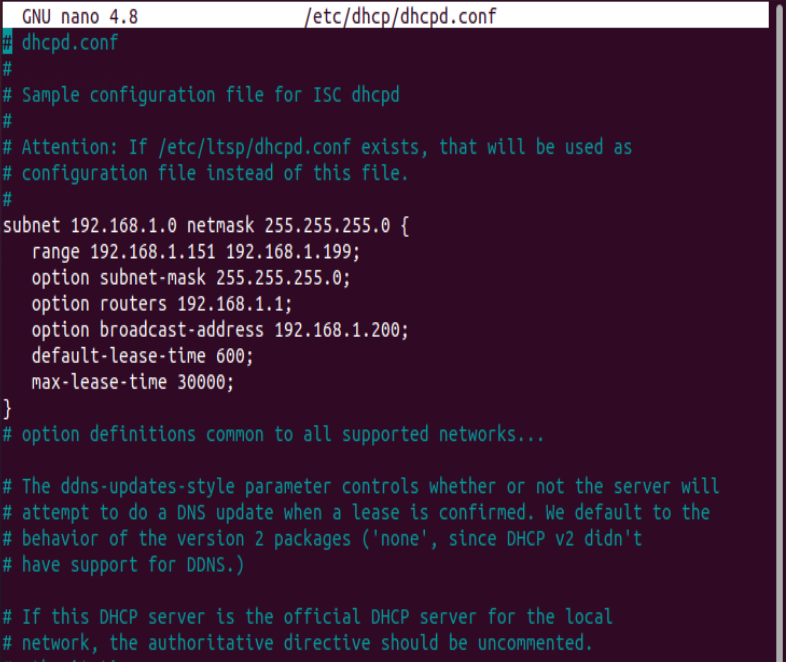
For the next step run the command **‘sudo nano /etc/default/isc-dhcp-server’** and enter the name of the interface into the line **‘INTERFACESv4=””’.**

**It should look like this:**



This will tell the DHCP server to listen to that interface.

Next run the command **‘sudo nano /etc/dhcp/dhcpd.conf’. enter the folling text shown in the screenshot blow using the ip addresses given:**



Save this file and exit.

The next part you will need to run **‘sudo nano /etc/netplan/99\_config.yaml’** so you can set up the new interface for the server.

Enter following as shown in the screenshot below:



Again, save and exit.

Next use the command **‘sudo netplan apply’** and then restart the server by using **‘sudo systemctl restart isc-dhcp-server’.**

Now, you need to open up your cloned vm and in the file **‘/etc/netplan/99\_config.yaml’** enter the following which includes the interface name:

Text

Description automatically generated

This means the address will be assigned using DHCP.

Now use the command **‘sudo netplan apply’** to apply the changes.

You should now reboot the clone whilst keeping the original VM open and when you run the ifconfig both VMs should have the IP addresses **192.168.1.150 and 192.168.1.151.**

Below shows what the **ifconfig** command should look like and a screenshot of them **pinging** each other to show its possible:

Graphical user interface, text, application, chat or text message

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

**NFS Server**

**Part 4:** this part shows how to set up an **NFS server**.

The first step is to install the nfs-kernel-server so we can create a shared directory with the client VM. To do this run the command **‘sudo apt install nfs-kernel-server’.**

Next run the command ‘**mkdir shared’** to create the directory you want to share with the client.

When using NFS it cant be associated with any other user account so we run the command **‘sudo chown -R nobody:nogroup shared/’** and ‘**sudo chmod 777 shared/’**

**Next you will need to export the directory by running the command ‘sudo nano /etc/exports’ and entering in the line ‘/home/<username>/shared <client IP>(rw,async, no\_subtree\_check) to the file. Below is an example:**

Text

Description automatically generated

Next you need to type in the command **‘sudo exportfs -a’** to read the new file. And then restart the server using **‘sudo systemctl restart nfs-kernel-server’**.

Next you need to run the command **‘sudo ufw allow from <client IP> to any port nfs’** and **‘sudo ufw enable’.** This will allow NFS connections into your system.

Next step on the client VM you need to run the command **‘sudo apt install nfs-common’.** This command will install the client software needed for this server. Next make the directory you want to be shared with the shared directory using **‘mkdir <directory name>’.**

**The final step will be to mount the folder to your new directory.**

Use the command **‘sudo mount <server IP>:home/<username>shared <directory name>’**

If you run the ls command you will see that they are shared.

Below are screenshots of an example shared directory with the client using the directory name **nfs\_dir** and **shared:**

Graphical user interface, text, application, chat or text message

Description automatically generated

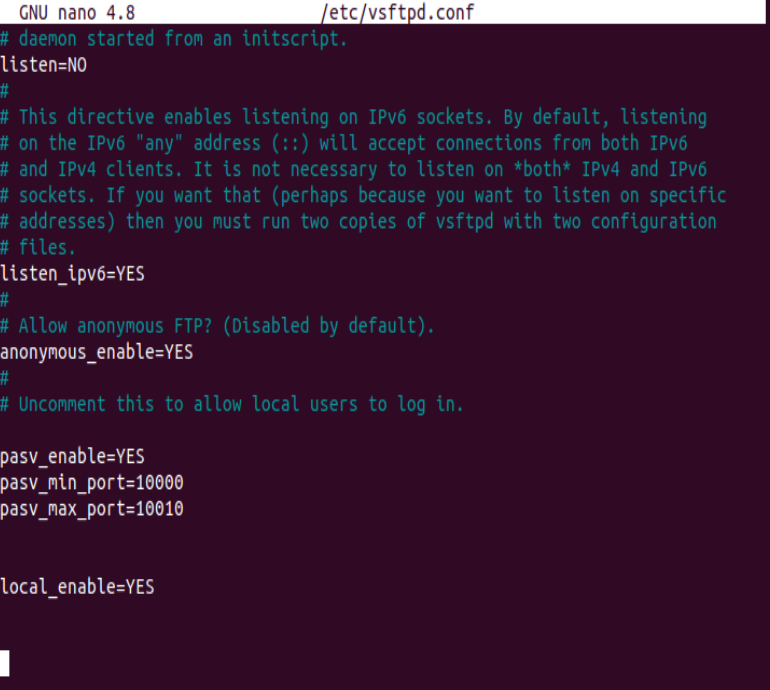
**FTP Server**

**Part 5:** This part will be setting up an FTP server. The aim of setting up the FTP server is to allow the server to accept passive connections and also to place files in the servers directory that can be shared to clients.

The first part of the setup is installing the server using the command **‘sudo apt install vsftpd’.**

Next we need to configure the server by editing the file stored in **‘/etc/vsftpd.conf’** by running the command **sudo nano /etc/vsftpd.conf’** .

The file should be the same as the screenshot below:



Once you have saved an exited run the command: **‘sudo systemctl restasrt vsftpd’**

Next we need allow FTP connections and open ports. Run the commands:

**‘sudo ufw allow 21/tcp’**

**‘Sudo ufw allow 10000:10010/tcp’**

The Final step will be to test the server by placing a file in the directory **/srv/ftp**. Once this is done you need to run a command to download by using the servers IP address and the name of the file you created on the client machine.

You should run the command in this format: **‘wget** [**ftp://server**](ftp://server) **IP address/my\_file.txt’**

Below is the commabd being run and the output you should get for the IP address **192.168.1.150** and the file **‘AssignmentFile.txt’. the IP address should normally be the enp0s8 interface address.**

Text

Description automatically generated

**Network Router**

**Part 6:** Setting up the server to operate as a Router.

Forst stwp is to enable packet forwarding on the file **/etc/sysctl.conf.** do this y adding the line **‘net.ipv4.ip\_forward=1’.**

Next we tell the server to forward form the enp0s8 interface to enp0s3 by using this command:

**‘sudo iptables -A FORWARD -I enp0s8 -o enp0s3 -j ACCEPT’**

We then do it the other way around:

**‘sudo iptables -A FORWARD -I enp0s3 -o enp0s8 -m state –state RELATED,ESTABLISHED -j ACCEPT’**

Next to enable NAT we run this command:

**‘Sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE’**

Finally, you will need to make the changes permanent by installing iptables-persistent and saving it using these commands:

‘Sudo apt install iptables-persistent’

**‘Sudo bash -c “iptables-save > /etc/iptables/rules.v4”’.**

In this next part we need to allow the packets by altering the firewall.

To do this you need to got into the directory **/etc/default/ufw** and change the line that says **‘DEFAULT\_FORWARD\_POLICY=”DROP”** to **ALLOW** instead of **DROP.**

Next open **/etc/ufw/sysctl.conf** and add **‘net/ipv4/ip\_forward=1’.**

Finally, disable and enable the firewall using **‘sudo ufw disable’** and **‘sudo ufw enable’.**

Make the changes permanent by running: **‘sudo netfilter-persistent save’**

The final part is setting up the client. You will need to modify two **.yaml** files.

First delete the **/etc/netplan/00-installer-config.yaml** file, then set up the **99\_config.yaml** file as shown in the screenshot below:

Text

Description automatically generated

Next run **‘sudo netplan apply’** to save both the files.

After all this close the client and go into settings and disable the network adaptor 1. This will get rid of the enp0s3 interface.

Now run the ifconfig command to check and you should see a screen like the one in the screenshot below.

Text

Description automatically generated

The final step is to ping google.ie to check if the internet is still working. Run the command **‘ping google.ie’**

Text

Description automatically generated