1. DNS
2. Install bind9 and check the installation
   1. Sudo apt install bind9
   2. Named -v
   3. Systemctl status named
3. Start the service if it isn’t running already
   1. Sudo systemctl start named
4. Text

   Description automatically generatedCreate /etc/bind/db.example.lan and enter the following

This is the configurations file for the server and

this is the db.example.lan.reverse for the reverse dig.

Text

Description automatically generated

named-checkzone example.com. /etc/bind/db.example.lan

the above checks that /etc/bind/db.example.lan constitute a valid zone file for   
the FQDN example.com. by checking for any typos.

After adding the new entries, we must add it to the named’s config file to tell it about our new zonefiles, which include the reverse dns file as well.

named-checkconf /etc/bind/named.conf is used to make sure that there is no typos and unrecoginised mistakes in the named.conf

Text

Description automatically generated

After updating the named.conf we must restart the service to make sure it has all the updates files.

Sudo systemctl restart named.

We should now be able to dig example.lan and also the reverse dig through the local host

Dig @127.0.0.1 example.lan

Dig @127.0.0.1 -x 10.20.30.41

2-DHCP Server for ip range 192.168.1.150 – 192.168.1.200

First we must install all the necessary modules and in this case its isc-dhcp-server

Second tell the dhcp server to listen on enp0s8 interface:

Text

Description automatically generated

Third we have to tell the server what IP addresses it should give a client, by making this file:

/etc/dhcp/dhcpd.conf

Text

Description automatically generated

This file contains info like the range. Subnet mask gateway and the broadcast address for the server.

Fourth we have to setup the network interface:

Text

Description automatically generated with medium confidence

This gives the host vm a static ip of 192.168.1.1

Now run this to apply the changes in configurations

Sudo netplan apply

Finally on the server run this to make sure all the changes have loaded:

sudo systemctl restart isc-dhcp-server

sudo systemctl status isc-dhcp-server

To check if the server is running properly.

CLONE

Now on the clone we must also configure its 99\_config

Text

Description automatically generated

In it we're saying we want the IP address to be assigned using DHCP, rather than giving a static   
address

Now run this to apply the changes in configurations

Sudo netplan apply

With dhcp-lease-list you should a record of the clone's DHCP transaction with the server.

1. NFS Server
2. Install nfs-kernel-server
3. Then create a directory where shared items will be placed with mkdir shared/
4. Change permissions for very wide access and no association with users
   1. sudo chown -R nobody:nogroup shared/
   2. sudo chmod 777 shared/
5. open /etc/exports and add this line for my setup



1. read in updated server and restart with:
   1. sudo exportfs -a
   2. sudo systemctl restart nfs-kernel-server
2. add rules to server firewall to allow nfs connections into the system
   1. sudo ufw allow from 192.168.1.150 to any port nfs
   2. sudo ufw enable
3. check the services is active and running correctly :
   1. sudo ufw status

Client/Clone

1. Install nfs-common which is the client software
2. Create a directory that will mount the shared directory from the server onto using nfs.
   1. Mkdir nfs\_dir
3. Mount the shared folder onto nfs\_dir with:
   1. sudo mount 192.168.1.1:/home/ integration/shared nfs\_dir

NFs should now be working, and this can be tested by creating a file in either shared/ on the server or nfs\_dir/ on the client machine and then you’ll find that there is a copy of it on the other machine.

4- FTP Server

1. Install the FTP daemon and then check its status using:
   1. Sudo apt install vsftpd
   2. Systemctl status vsftpd
2. Configure the server to accept anonymous and passive connections in /etc/vsftpd.conf
   1. Change anonymous\_enable =NO to anonymous\_enable =YES
   2. Then add:

pasv\_enable=YES  
pasv\_min\_port=10000  
pasv\_max\_port=10010

Text

Description automatically generated

1. Restart the server to make changes take effect:
   1. Sudo systemctl restart vsftpd
2. Change the firewall to allow connections on port 21 and open ports 1000 to 10010
   1. Sudo ufw allow 21/tcp
   2. Sudo ufw allow 10000:10010/tcp
3. Change the directory which holds the ftp files to be accessed
   1. Sudo usermod -d /ftp\_dir ftp

Clone/Client

Run this command to download a file from the host:

wget ftp://server IP address/my\_file.txt

Example:

Text

Description automatically generated

5- Router

1. Open the /etc/sysctl.conf and add the following line
   1. net.ipv4.ip\_forward=1



1. Tell the server how forwarding should work. Forward packets from enp0s8 to enp0s3
   1. sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -j ACCEPT
2. Forward it to the other direction, from enp0s3 to enp0s8:
   1. sudo iptables -A FORWARD -i enp0s3 -o enp0s8 -m state --state RELATED,ESTABLISHED -j ACCEPT
3. Enable Network Translation:
   1. sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
4. install iptables-persistent to save the changes:
   1. sudo apt install iptables-persistent
   2. sudo bash -c "iptables-save > /etc/iptables/rules.v4"
5. Now open /etc/default/ufw to edit the firewall settings as it might block masquerade packets
   1. Change the “DROP” to “ACCEPT” on DEFAULT\_FORWARD\_POLICY=”DROP”

A screenshot of a computer

Description automatically generated with medium confidence

1. Open /etc/ufw/sysctl.conf and add the following:
   1. net/ipv4/ip\_forward=1



1. Finally Restart the firewall and save the changes:
   1. Sudo ufw disable
   2. Sudo ufw enable
   3. Sudo netfilter-persistent save

Clone/Client

1. Move the /etc/netplan/00-installer-config.yaml as it isn’t needed anymore.
2. Edit the /etc/netplan/99\_config.yaml by adding your address

Shape, rectangle

Description automatically generated

1. Save the changes made:
   1. Sudo netplan apply
2. Shutdown the client, disable adaptor 1 then reboot.

TEST

$ping 8.8.8.8