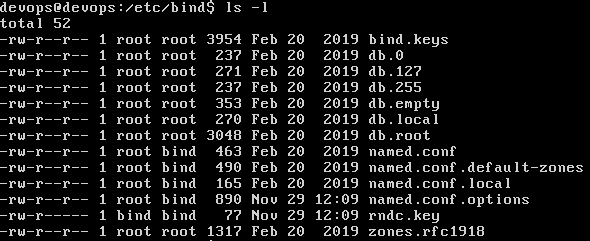
**Systems Integration Assignment**

***Author: Jamie Walsh***

**Part 1: DNS server for the domain example.lan with forward and reverse lookup**

***Step 1: Installing the DNS Server***

1. Before setting up the DNS cache server, we update the default repositories and do a system upgrade. This is done using the following command: sudo apt-get update && sudo apt-get upgrade -y
2. Next, install the DNS Packages bind and dnsutils using the following command: sudo apt-get install bind9 dnsutils -y
3. Once you have installed DNS change your directory to etc/bind/ using the cd command. cd etc/bind/
4. When in this directory, list the directory contents by using the following command: ls -l

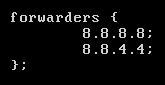


The DNS Packages have successfully been installed and the relevant files are in the directory ready to be configured.

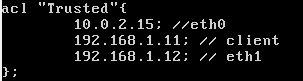
***Step 2: Setup the DNS Cache Server***

Still within the etc/bind directory, open and edit the “named.conf.options” file using the nano file editor. This is done using the following command: sudo nano named.conf.options

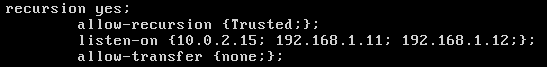
1. Now, in this file locate “forwarders”, uncomment out the forwarders section and change the 0.0.0.0 to the public google DNS IP addresses 8.8.8.8 and 8.8.4.4 or your own ISP router’s IP address. Then save your changes and exit the file.



1. Also, while in the options file, we can add the access control list (acl). The acl clause allows fine-grained control over what hosts or users may perform what operations on the name server. I added the eth0 IP address, and the eth1 IP address of both the client and the server.

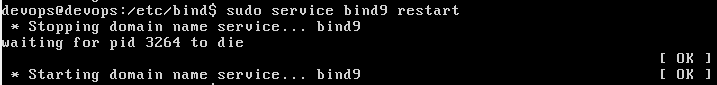


1. Next, we want to allow recursion to these trusted IP addresses. We do this by adding the text seen in the image below to the same file.

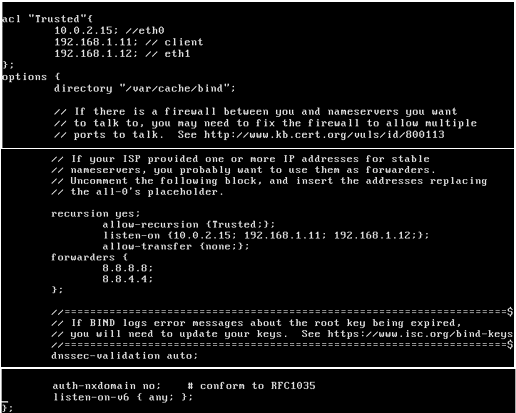


With recursive requests, your server will attempt to find the website in question in its local cache. If it can’t find an answer in the local cache, then it will query other DNS servers on your behalf until it finds the address.

1. After making all these changes to the options file, you need to restart the bind daemon. Every time you make changes to the bind files you need to restart the server. This is done using the following command: sudo service bind9 restart. You know this worked when you get two “[ OK ]” messages.



Once the bind service successfully restarts, this means your named.conf.options file is properly configured. The file should look something like the screenshot below at this stage.

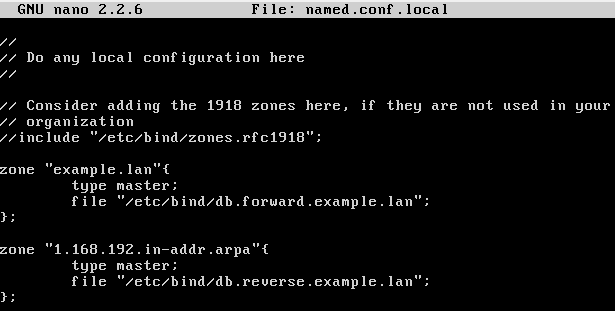


***Step 3: Setup Master DNS Server***

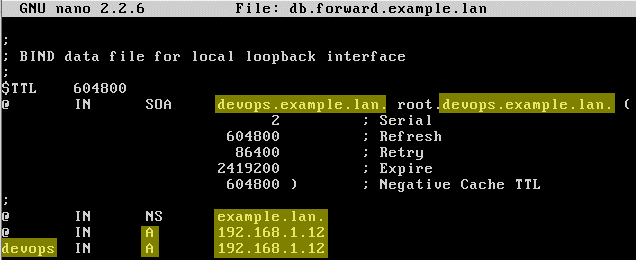
1. Now, we create a master DNS server. To do this we first open the “named.conf.local” file using sudo nano named.conf.local.
2. Add a new zone for the DNS-Master entry in this file. Name the zone the domain name “example.lan”. set the type to master and the file to db.forward.example.lan.



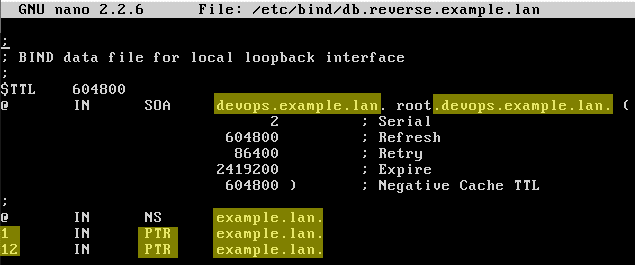
1. The reason I named the forward look-ups file “db.forward.example.lan” is because we also want to support reverse look-ups. To add reverse look-up, enter a second zone below the first one and set the name to “1.168.192.in-addr.arpa”. You can name the look-up file, “db.reverse.example.lan”. Your local file should look like the screenshot below once complete.



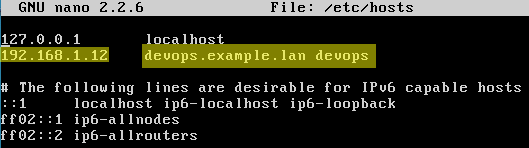
1. To create the zone file (forward look-up) make a copy of the db.local file using the following command: sudo cp db.local db.forward.example.lan
2. Open the newly created file using the sudo nano db.forward.example.lan command.
3. Edit the file and change localhost to the fully qualified domain name (FQDN), which in our case is example.lan, and leave a dot (”.”) at the end.
4. At the bottom, change 127.0.0.1 to the nameserver’s IP address. Then copy and paste that line to create a new “A” client record below the existing one and set the name to the client’s hostname (devops in our case). Your forward file should resemble the screenshot below.



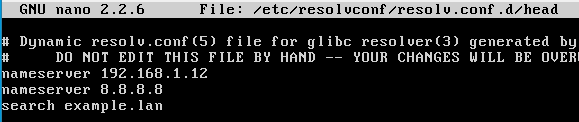
1. To create the reverse look-up zone file, repeat step 4, 5 and 6 changing db.forward.example.lan to db. reverse.example.lan.
2. For each “A” client record you configure in db.forward.example.lan, you need to create a PTR client record in db.reverse.example.lan. Save and exit the file. Your reverse file should look like the image below at this stage.



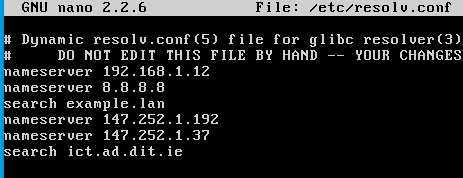
1. At this stage, it is important to check the host ip address in order to ensure that the eth1 inet address of the server is the same as the host. This is done using the command: sudo nano /etc/hosts
2. Within the /etc/hosts file, confirm that the correct ip address is being used (in our case 192.168.1.12) and change the hostname to “devops.example.lan devops”.



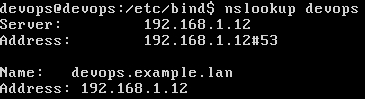
1. Now, you should look at the resolvconf file. This is a file that configures the system's Domain Name System (DNS) resolver. You cannot edit this file directly but you can edit the head file which implements the changes to resolvconf. Open the file using: sudo nano /etc/resolvconf/resolv.conf.d/head
2. Add the nameserver with your DNS server’s IP address and Google’s DNS’s IP for the backup address. Also add the line, “search example.lan” as shown below.



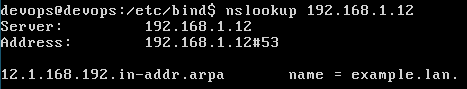
1. Once you have saved your changes and exited the file, run the command sudo resolvconf -u to configure these changes.
2. You can check if your changes were successful by looking into the resolv.conf file using running the following command: sudo nano /etc/resolve.conf This should display your changes.



1. Now the configuration is complete. To test DNS forward lookup, use the command nslookup devops. You should receive the server and address information with no errors. You can also try the dig command.

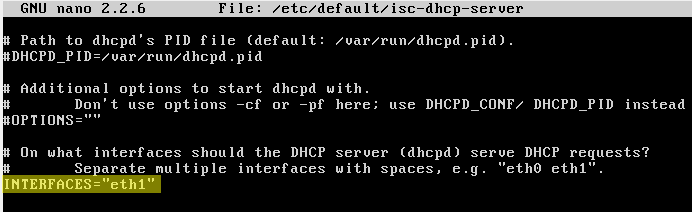


1. To test DNS reverse lookup, use the command nslookup 192.168.1.12. You should receive the server and address information with no errors.

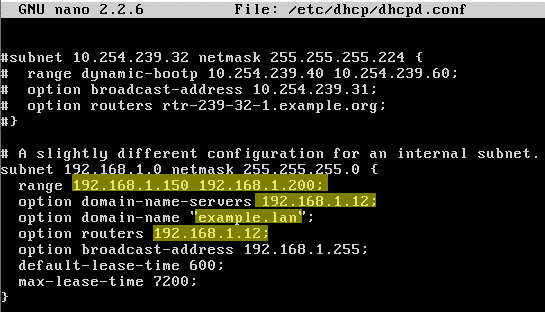


**Part 2: DHCP server to clients in the range 192.168.1.150 – 200**

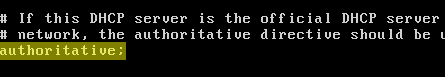
1. Sudo apt-get install isc-dhcp-server to download the dhcp server on to your ubuntu server.
2. Next open the isc-dhcp-server file using the following command: Sudo nano /etc/default/isc-dhcp-server.
3. In this file, scroll down to the bottom of the file and locate “INTERFACES”. Add eth1 to the interfaces list as shown in the screenshot below. Save and exit the file.



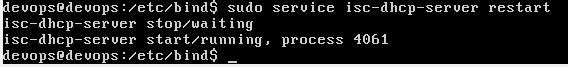
1. Then open the dhcpd.conf file using: Sudo nano etc/dhcp/dhcpd.conf. In there, change the internal subnet configuration to set up dhcp server, range, subnet, etc.



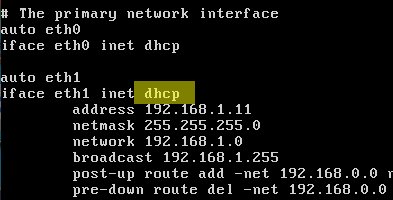
1. Also, in the same file, locate and uncomment “authoritative”.



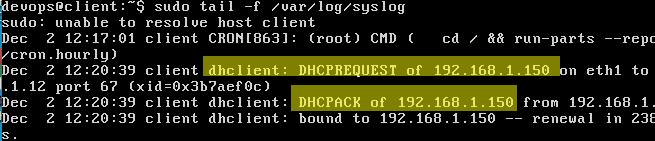
1. Sudo service isc-dhcp-server restart to restart the server in order to save the changes.



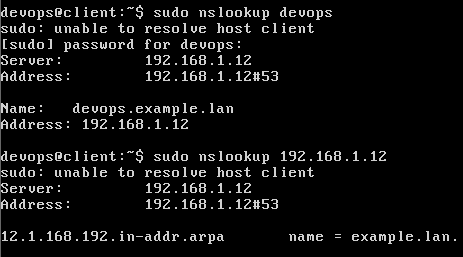
1. Ensure that eth1 on your client is using dhcp instead of static. Open the interfaces file using: sudo nano /etc/network/interfaces and check eth1.



1. Now run the sudo reboot command to save all the changes made on the client.
2. Test by using ifconfig eth1 on the client. The IP Address should be 192.168.1.150. you can also use sudo tail -f /var/log/syslog to see the end of the log files on startup of the client. This will tell you if DHCP was used to set the IP.



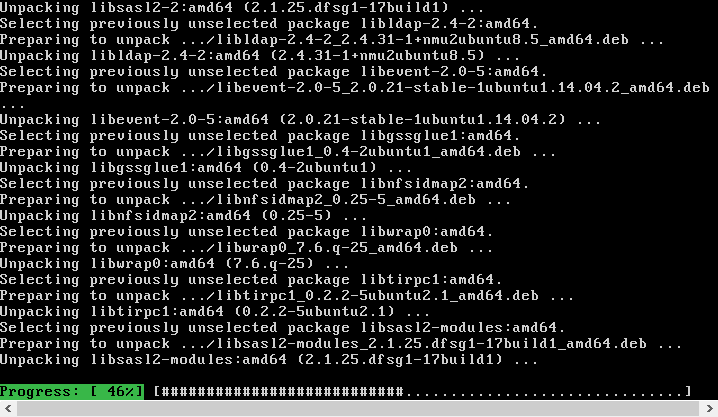
1. Once your client’s IP address is changed to 192.168.1.150, it is important to remember to change the named.conf.options file in the server to reflect these changes. Then you can perform DNS on the client too.



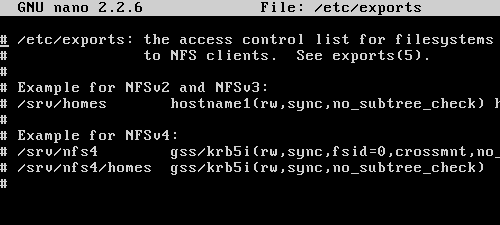
**Part 3: NFS server to share /home/myshare**

***Step 1: Setup NFS on Server***

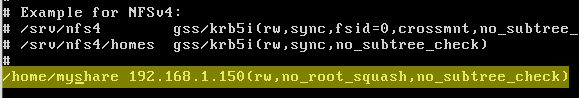
1. First, update the install the NFS server using the following command: sudo apt install nfs-kernel-server
2. You will be prompted to allocate the required memory, reply with y [yes]. The package will begin to download as shown below.



1. Make the directory you want to share to other users. In our case the command is: sudo mkdir /home/myshare
2. Open the /etc/exports file. This is the main config file for NFS. Command: sudo nano /etc/exports



1. Add a share directory to the exports file using the highlighted command in the screenshot below. The IP address should be your client’s IP. Save your changes and exit the file.



1. Now start the service using the following command: sudo /etc/init.d/nfs-kernel-server start



1. You can check the NFS share status using sudo in the screenshot below. Your shared files’ directories and IP addresses should be displayed.

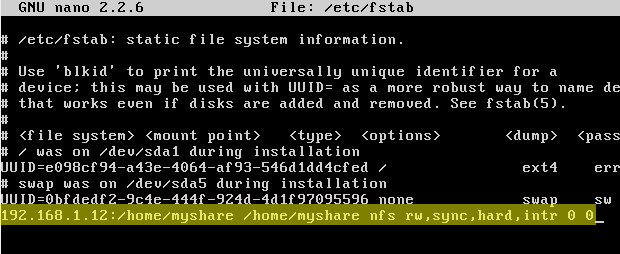


***Step 2: Setup NFS on Client***

1. Install NFS client and dependencies using the following command: sudo apt-get install nfs-common rpcbind
2. Create a local directory on the client. Command: Sudo mkdir /home/myshare
3. You now mount the remote share directory on to your local directory using the command below.



1. To make the mount permanent add the following line in /etc/fstab



1. Run the mount command to check the mounted shared directory. This shows that the local directory is now a remote NFS directory.



**Part 4: FTP server**

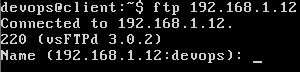
***Step 1: Installing the FTP Server***

1. Update the system packages Command: sudo apt-get update
2. Install the vsftpd package. Sudo apt-get install vsftpd

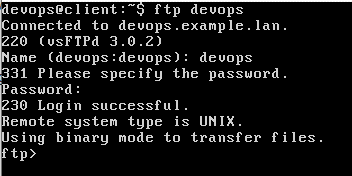


***Step 2: Test VsFTP Server***

1. Install FTP on your client using sudo apt-get install ftp
2. Attempt to connect to the FTP server from the client. This is done using the following command: ftp 192.168.1.12

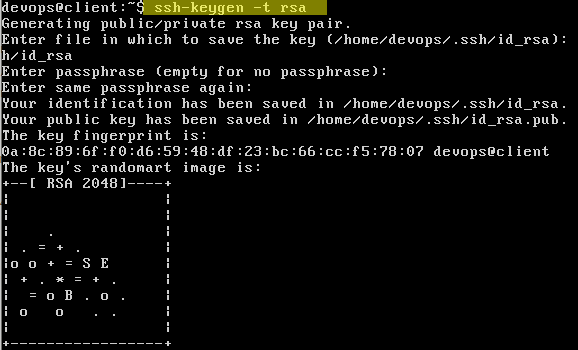


1. Connect to the ftp server using the hostname to prove that DNS is working. Command: ftp devops

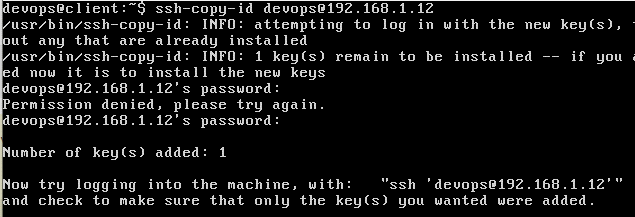


**Extra: SSH**

1. Install openssh-server on the client and server.
2. First you need to create the key pair using ssh-keygen -t rsa on your client machine.



1. Use the following command to copy the public key to your server: Ssh-copy-id devops@192.168.1.12



1. Using the ssh command, try ssh to the server. You shoud be able to login without a password thanks to the keys.



**Extra: NTP**

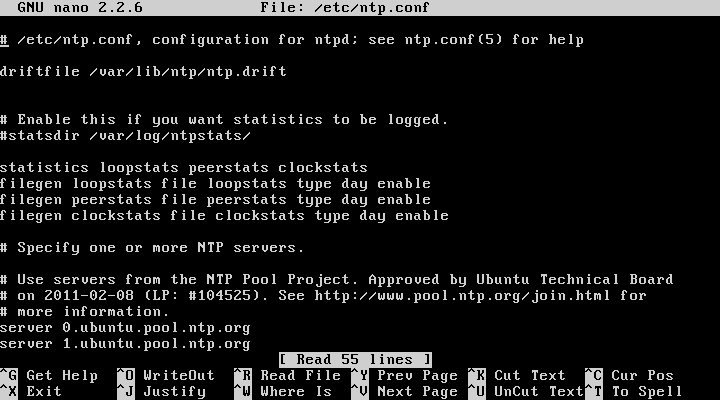
1. Update and install the packages as usual with the commands below.

sudo apt-get update

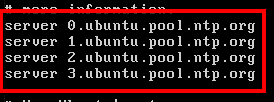
sudo apt-get dist-upgrade

sudo apt-get install ntp ntpdate

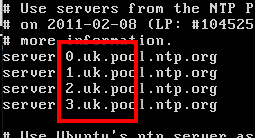
1. Open the /etc/ntp.conf file using the nano command.



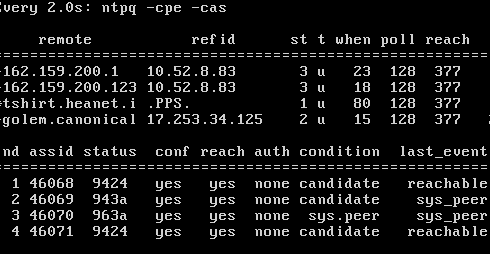
1. Locate the pool servers within this file. As you need 3 servers to form a quorum for ntp, it is good to maintain at least 4 servers.



1. Change “ubuntu” to a “uk”. Save and exit the file.



1. From here, go to the client and install NTP and ntpdate too.
2. You can then run the following command to view NTP in action.



1. Stopping, starting or restarting the NTP server can be achieved by using the following NTP commands. sudo service NTP start, sudo service NTP stop, sudo service NTP restart