

*System Admin*

*Training Assignments*

|  |  |
| --- | --- |
| **Program Code** |  |
| **Issue/Revision** | **x/y** |
| **Effective date** | **04/Aug /2023** |

**Assignment Day 06. System Software - Day 1**

**Mục Lục:**

[1. Cài đặt và cấu hình dịch vụ DNS Server 3](#_Toc1683636514)

[2. Cài đặt và cấu hình dịch vụ NTP Server 6](#_Toc551488244)

[3. Cài đặt và cấu hình dịch vụ DHCP Server 8](#_Toc1875854875)

[4. Cài đặt và cấu hình dịch vụ FTP Server 8](#_Toc1746202608)

# Cài đặt và cấu hình dịch vụ DNS Server

Ubuntu 20.04 OS:

**Install DNS Server**

First, update the repository index.

sudo apt update

sudo apt install -y bind9 bind9-utilsCOPY

**Creating DNS Zones**

The /etc/bind/ is the configuration directory for the DNS server that holds configuration files and zone files. The global configuration file for DNS server is /etc/bind/named.conf.

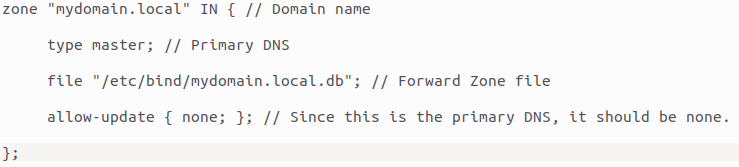
For the zone creations, use the /etc/bind/named.conf.local file instead of the global configuration file.

Let us begin by creating DNS zones for your domain.

sudo vim /etc/bind/named.conf.local

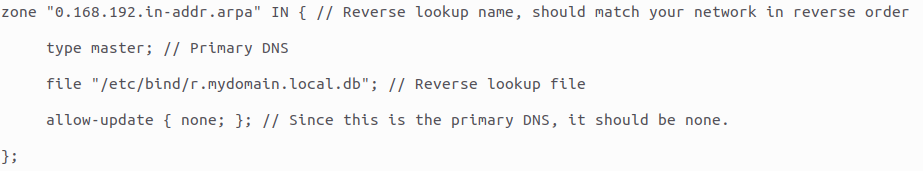
**Creating Forward Zone**

The following is the forward zone entry for the mydomain.local domain in the named.conf.local file. This forward zone translates a fully qualified domain name (FQDN) into an IP address.



**Creating Reverse Zone**

The following is for the reverse name resolution zone in the named.conf.local file. This forward zone translates an IP address into a fully qualified domain name (FQDN).



**Creating Zone Files**

Once you have created DNS zones, create DNS zone files for the forward and reverse zones.

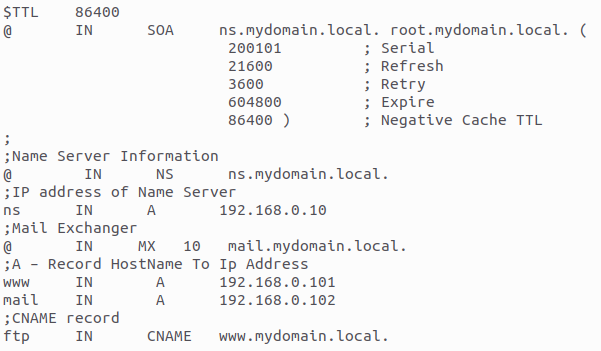
**Creating Forward Zone File**

Create a zone file /etc/bind/mydomain.local.db for the forward zone.

sudo vim /etc/bind/imydomain.local.db

Record types in the zone file,

SOA – Start of Authority  
NS – Name Server  
A – A record  
MX – Mail for Exchange  
CN – Canonical Name



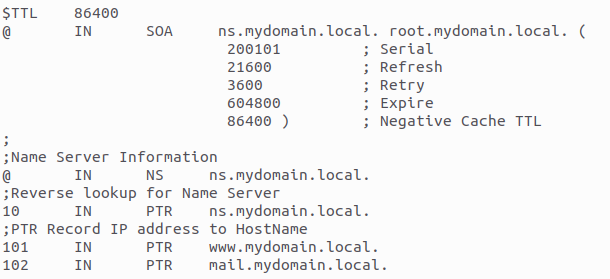
**Creating Reverse Zone File**

Create a zone file /etc/bind/r.mydomain.local.db for the reverse zone.

sudo nano /etc/bind/r.mydomain.local.db

Update the content as shown below.

PTR – Pointer  
SOA – Start of Authority



**Validating DNS Syntax**

Use the named-checkconf command to check the syntax of DNS configuration files for any errors.

sudo named-checkconf

Command will return to the shell if there are no errors.

Also, you can use named-checkzone to check the syntax errors in zone files.

**Validating Forward Zone**

sudo named-checkzone mydomain.local /etc/bind/mydomain.local.db

Output:

zone mydomain.local/IN: loaded serial 200101  
OK

**Validating Reverse Zone**

sudo named-checkzone 0.168.192.in-addr.arpa /etc/bind/r.mydomain.local.db

Output:

zone 0.168.192.in-addr.arpa/IN: loaded serial 200101  
OK

Finally, reload both the configuration file and zones. You may also use the same command whenever you change zone and zone files.

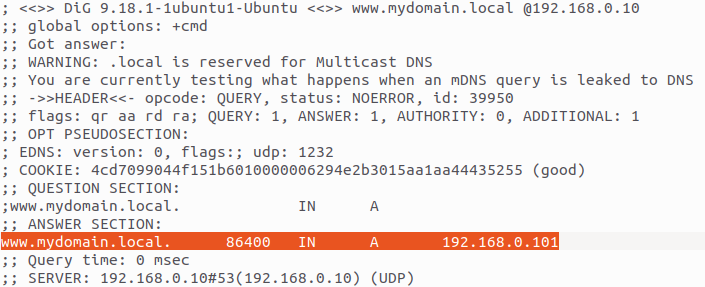
sudo rndc reload

**Verify DNS Server**

Use the dig command to verify the DNS server by looking up records.

dig [www.mydomain.local](http://www.mydomain) @192.168.0.10

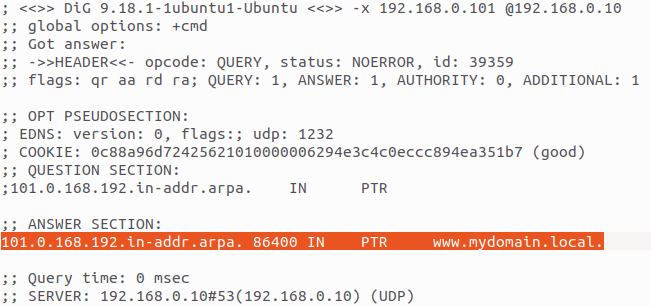
Output:



Confirm the reverse lookup with the dig command.

dig -x 192.168.0.101 @192.168.0.10

Output:



# Cài đặt và cấu hình dịch vụ NTP Server

Ubuntu 20.04 OS:

**Install NTP server**

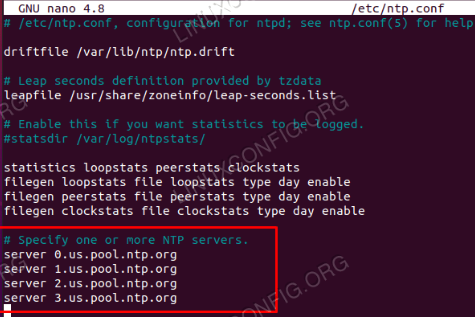
1. To begin, we need to install NTP server. You can do so by opening a terminal and entering the following command:

$ sudo apt install ntp

**Configure NTP server**

1. The NTP server comes pre-configured with some server pools already, which you can see inside the /etc/ntp.conf file.

$ vim /etc/ntp.conf



1. Once you’ve made these changes, save and exit the configuration file. Restart the NTP service for the changes to take effect:

$ sudo systemctl restart ntp

1. Check on the status of the NTP service at any time with this command:

$ sudo systemctl status ntp

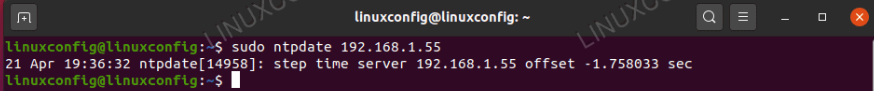
**NTP client configuration**

1. First, we need to install the ntpdate package. We can use this to verify connectivity between the client and the NTP time server we created.

$ sudo apt install ntpdate

1. Next, let’s attempt to mantually sync our system time with the NTP server. Type the following command, substituting your NTP server’s IP address or hostname where appropriate:

$ sudo ntpdate 192.168.1.55



1. That seems to be working as we’d expect. Next, be sure to disable Ubuntu’s default timesyncd service, as this will conflict with our attempts to synchronize with the NTP server.

$ sudo timedatectl set-ntp off

1. Now, we need to install the NTP daemon on our client system so we can configure it to pull the time from our NTP server that we set up earlier.

$ sudo apt install ntp

1. We only need to add a single line to our ntp.conf file, and we can do that very easily with a single command. Just make sure to replace the IP address below with either the hostname or the IP address of your NTP server.

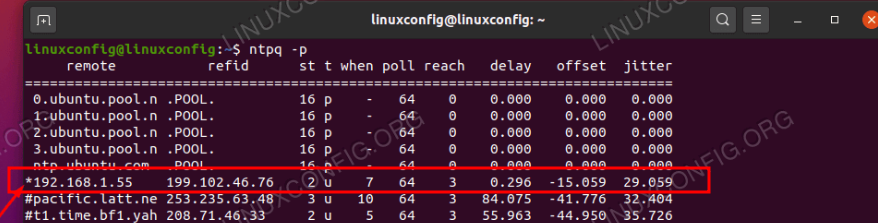
$ sudo bash -c "echo server 192.168.1.55 prefer iburst >> /etc/ntp.conf"

1. Then, restart the NTP daemon:

$ sudo systemctl restart ntp

1. Lastly, use the ntpq command to list the NTP time synchronization queue:

$ ntpq -p



# 3. Cài đặt và cấu hình dịch vụ DHCP Server

**Install isc-dhcp-server**

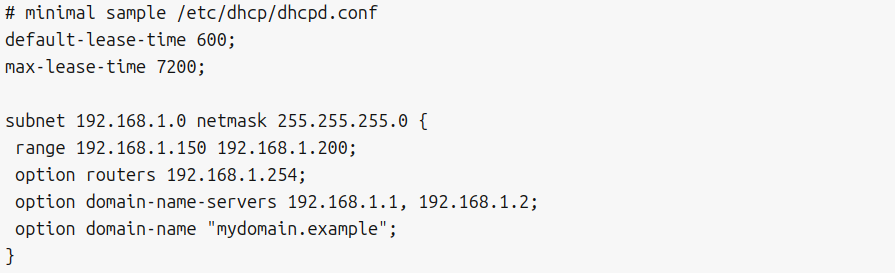
At a terminal prompt, enter the following command to install isc-dhcp-server:

sudo apt install isc-dhcp-server

**Configure isc-dhcp-server**

You will probably need to change the default configuration by editing /etc/dhcp/dhcpd.conf to suit your needs and particular configuration.

Most commonly, what you want to do is assign an IP address randomly. This can be done with /etc/dhcp/dhcpd.conf settings as follows:



This will result in the DHCP server giving clients an IP address from the range 192.168.1.150 - 192.168.1.200. It will lease an IP address for 600 seconds if the client doesn’t ask for a specific time frame. Otherwise the maximum (allowed) lease will be 7200 seconds. The server will also “advise” the client to use 192.168.1.254 as the default-gateway and 192.168.1.1 and 192.168.1.2 as its DNS servers.

You also may need to edit /etc/default/isc-dhcp-server to specify the interfaces dhcpd should listen to.

INTERFACESv4="eth4"

After changing the config files you need to restart the dhcpd service:

sudo systemctl restart isc-dhcp-server.service

# Cài đặt và cấu hình dịch vụ FTP Server

**VSFTPD Installation**

Install VSFPTD on Ubuntu 20.04, along with some supporting packages:

1. Update your system’s packages:

sudo apt update

1. Install the VSFTPD server, the FTP command line client, and the UFW firewall. The FTP command line client is used in this guide to issue local test connections to the VSFTPD server:

sudo apt-get install vsftpd ftp -y

1. Set VSFTPD to start whenever your server boots:

sudo systemctl enable vsftpd

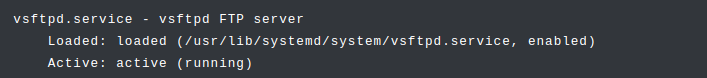
1. Launch VSFPTD:

sudo systemctl start vsftpd

1. Verify that VSFTPD is running properly after this installation:

sudo systemctl status vsftpd

1. You should see output similar to:



**Create an FTP User**

To see VSFTPD in action–a kind of “Hello, world” for FTP–create a special-purpose user on your server:

1. Create a Linux user named ftp\_client:

sudo useradd -m ftp\_client

1. Set the password for your new user:

sudo passwd ftp\_client

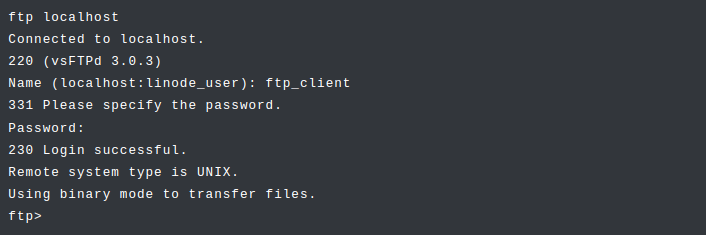
1. Create an example text file under the home directory of the new ftp\_client user:

sudo -u ftp\_client sh -c 'echo "This is the content in the file." > /home/ftp\_client/testfile.txt'

1. Open an FTP connection to the VSFTPD server running on localhost. This syntax is similar to connections you would make from remote systems, which is demonstrated later in this guide:

ftp localhost

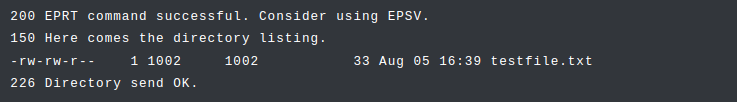
1. You are prompted for your FTP username (‘ftp\_client’), and then prompted for this user’s password (set in step 2 of this section). After entering this information successfully, an ftp> command prompt appears:



1. Verify that the sample file testfile.txt in the ftp\_client user’s home directory is visible from the FTP connection:

ls /home/ftp\_client

1. The output resembles:

Y

1. Close the ftp client with the exit or quit commands:
2. exit  
   221 Goodbye.
3. You have verified that your VSFTPD accepts connections. The next sections show where you can configure more sophisticated account management, encryption, and security restrictions:

**How to Restart VSFTPD**

VSFTPD is restarted via systemctl:

sudo systemctl restart vsftpd

When VSFTPD starts or restarts, it reads from the current configuration files for the service, which are detailed in the next section.

**VSFTPD’s Configuration File**

In Ubuntu and other common distributions, VSFTPD’s configuration is located in /etc/vsftpd.conf. When updating the configuration file, follow these steps:

1. *(Recommended)* Back up the current configuration by making a copy:

sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.bak

1. Edit the /etc/vsftpd.conf configuration file in your preferred text editor.
2. Restart VSFTPD to activate the changes:

sudo systemctl restart vsftpd

As its name promises, one of VSFTPD’s goals is security. It offers a wide range of settings to help match a range of security and business requirements. The [official manual page](http://vsftpd.beasts.org/vsftpd_conf.html) outlines all configuration options available. The next section introduces some relevant permissions.

**VSFTPD User Permissions**

To make VSFTPD useful for some real-world use-cases, you can adjust some of the default options set in vsftpd.conf:

1. Open /etc/vsftpd.conf in your text editor.
2. Locate the following recommended options within the file. Some may be commented out. If an option is commented out, remove the comment (by removing the # prefix at the beginning of the line). Some options may not be listed in the file. For these options, add a new line with the option. When finished making changes, save the file.

* anonymous\_enable: Set this option to NO (anonymous\_enable=NO). This blocks anonymous logins to the FTP server.
* local\_enable: Set this option to YES (local\_enable=YES). This allows you to log in as the users specified in your system’s /etc/passwd file.
* write\_enable: Set this option to YES (write\_enable=YES). This allows you to make changes to the filesystem via FTP, including uploading files.

1. Restart VSFTPD to activate these changes:

systemctl restart vsftpd

**VSFTPD Log File**

VSFTPD logs its actions. The default location of the log file is /var/log/vsftpd.log. The configuration attribute xferlog\_file controls this location. View its content from time to time to understand the information the logfile preserves:

sudo more /var/log/vsftpd.log

**Downloading with VSFTPD**

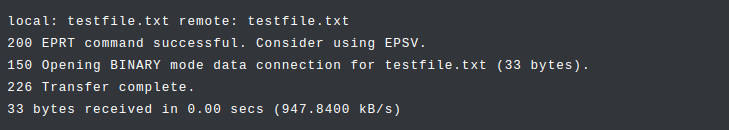
1. On the server, open an FTP connection to localhost:

ftp localhost

1. Enter the ftp\_client username and password when prompted.
2. At the FTP command prompt, change directory to the ftp\_client home directory:

cd /home/ftp\_client  
250 Directory successfully changed.

1. Use the get command to retrieve the test file that was created in the [Create an FTP User](https://www.linode.com/docs/guides/vsftpd-on-ubuntu-2004-installation-and-configuration/#create-an-ftp-user) section:
2. get testfile.txt



1. Exit the FTP session:

exit

1. Observe that the file is now present in your original user’s home directory:

ls –l



**Uploading with VSFTPD**

1. Create a text file in your system’s /tmp directory:

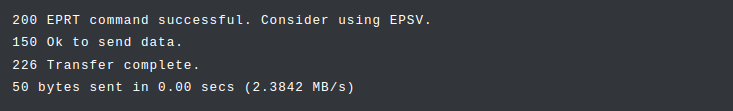
cd /tmp  
 echo "This is sample content for uploading through FTP." > testfile2.txt

1. Open an FTP connection to localhost. Enter the ftp\_client username and password when prompted:

ftp localhost

1. Within the FTP session, upload the file created in step 1 by using the put command:

put testfile2.txt



1. Exit the FTP session:

quit

1. Verify that the sample file testfile2.txt was uploaded to the ftp\_client home directory via FTP:

ls /home/ftp\_client

1. The output should resemble:



1. testfile.txt testfile2.txt

**Download/Upload using FileZilla Client**

