Curtin College | in association with Curtin University

CS2000 – Laboratory 02

Install and Configure DNS, FTP and Web Server on Linux Server

Trimester 3, 2021

Activate network services On Ubuntu Server

Task1: Configuring FTP Server

File Transfer Protocol (FTP) is a protocol for transferring files. It facilitates these transfers with proper integrity, efficiency and security. A client computer connects to an FTP server, where they can supply logon credentials, and then be granted access to retrieve files on the server.

Access to an FTP server can be managed in two ways:

- Anonymous
- Authenticated

In the Anonymous mode, remote clients can access the FTP server by using the default user account. In the Authenticated mode a user must have an account and a password. User access to the FTP server directories and files is dependent on the permissions defined for the account used at login. As a general rule, the FTP daemon will hide the root directory of the FTP server and change it to the FTP Home directory. This hides the rest of the file system from remote sessions.

In this part of the lab you are required to activate FTP on your server, then from your client side connect to it and perform the file transfers.

THIS IS THE COMMAND/INSTRUCTION **(THIS IS THE EXPLANATION)**

1. **Download and install vsftpd**
   sudo apt install vsftpd
2. **Configure the firewall to allow FTP and enable it**
   sudo ufw enable
   sudo ufw allow 20/tcp (port required for FTP to function)
   sudo ufw allow 21/tcp (port required for FTP to function)
   sudo ufw allow 40000:50000/tcp
3. **Create a user for FTP on your server**

sudo adduser *ftpuser* (adds a user to the server named ftpuser)

a. **Creating a user will automatically create a home directory /home/ftpuser**

4. **Edit the vsftpd configuration file to allow the new user to write**
   sudo nano /etc/vsftpd.conf (opens the VSFTPD configuration file with admin permissions)
   uncomment write_enable=YES (allows the user who connects via FTP to write/edit)
   pasv_min_port=40000 (optionally configure passive mode ports)
   pasv_max_port=50000 (optionally configure passive mode prots)
5. **Create a test file on your server.**
   Create test file in home directory
6. **Go back to your host machine (Windows 10) and use it as FTP client to connect to your Virtual FTP server.**
7. **Make a ftp connection from your client to your FTP server**
   a. **You can use command line or FTP client program**
   b. **Download and install FileZilla from: https://filezilla-project.org/index.php**
   c. **In Host window type *ftp://IPaddress of your virtual machine***
   Note: to check server IP address use *ifconfig* command.
   d. **Use your FTP user username and password that you created.**

Note: if there is a logging problem restart Ubuntu VM or the VSFTPD service

8. **Download the test file.**
9. **Upload a file from your client to your server.**
10. **Make your FTP server secure (limit users to an ftp files directory within their home directory)**
    a. **Limit users to their home directory**
       sudo mkdir /home/ftpuser/ftp (create directory ftp within /home/ftp)
       sudo chown nobody:nogroup /home/ftpuser/ftp (change the owner of the directory to nobody)
       sudo chmod a-w /home/ftpuser/ftp (remove write permissions for everone)
       sudo mkdir /home/ftpuser/ftp/files (create files directory)
       sudo chown ftpuser:ftpuser /home/ftpuser/ftp/files (change the owner to ftpuser)

       edit vsftpd.conf as below
       uncomment chroot_local_user=YES (prevent user from browsing outside their home directory)

       (below two commands tells vsftpd that the local_root is the /ftp folder we created earlier)
       user_sub_token=$USER
       local_root=/home/$USER/ftp
11. **Example of step by step configuration (required steps: 1, 2, 3, 4.2, and 5):**
    https://devanswers.co/installing-ftp-server-vsftpd-ubuntu-18-04/
12. **Once reconnected, attempt to upload a test file to the root directory, and one to the files directory.**
    The root directory should fail, the upload to the files directory should succeed.

Task2- Web Server

A Web server delivers Internet Web pages to client computers. The client runs a Web browser, which makes a request for an HTML web page from the server. The server services that request, and sends the desired page to the client's Web browser, where it is displayed.

Set up a Web Service on Ubuntu server.

1. **Install apache2 web server**
   sudo apt install apache2 (install apache2 web server)
   sudo ufw allow Apache (allow apache web server service through the firewall)

   sudo ufw enable (enable firewall)
   sudo ufw status (check firewall status)
   sudo systemctl status apache2 (check apache web server status)

   Test on your host machine by browsing to http://<server-ip>
   Use ifconfig command to check your servers IP address
2. **Following page is a good reference. Step 4 is optional and won't work as we don't have an online domain name. However, it is recommended for learning skills.** https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-ubuntu-18-04-quickstart

3. **Find the path to the home page and change the home page**
   sudo nano /var/www/html/index.html (this is where the home page is stored)
   a. **Add at least one more page to your website and view it in your host machines browser**
      sudo cp /var/www/html/index.html /var/www/html/index2.html (copies the original index.html file and creates a copy of it called index2.html)
      htttp://<server-ip>\index2.html (loads the secondary index2.html page)

Task3- DNS Server

A Domain Name Service (DNS) server keeps a database of tables that translate fully qualified Internet Domain names to their respective IP address. This enables you to refer to Internet servers by name, such as www.google.com rather than by IP address, such as 216.58.199.36.

Activate DNS server on your machine.

1. **What is the program maintaining a name server in linux?**
   BIND
2. **Install DNS server on your machine**
   sudo apt install bind9 (installs BIND)
   sudo apt install dnsutils (installs DNSUTILS which can help you test/troubleshoot)
3. **Write the steps for configuring DNS server (you are required to configure caching nameserver and Primary Master)**
   Caching
   sudo nano /etc/bind/named.conf.options (open named.conf.options file with admin perms)
   add 8.8.8.8 under forwarders (tells the caching server to lookup addresses with the 8.8.8.8 DNS server – this is Googles DNS server)
   sudo systemctl restart bind9.service (restart the BIND service)

   Master
   sudo nano /etc/bind/named.conf.local (open named.conf.local with admin perms)

```
zone "example.com" {
    type master;
    file "/etc/bind/db.example.com";
};
```

sudo cp /etc/bind/db.local /etc/bind/db.example.com

Edit the new zone file `/etc/bind/db.example.com` and change `localhost.` to the FQDN of your server, leaving the additional `.` at the end. Change `127.0.0.1` to the nameserver's IP Address and `root.localhost` to a valid email address, but with a `.` instead of the

usual `@` symbol, again leaving the `.` at the end. Change the comment to indicate the domain that this file is for.

```
;
; BIND data file for example.com
;
$TTL    604800
@       IN      SOA     example.com. root.example.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL

@       IN      NS      ns.example.com.
@       IN      A       192.168.1.10
@       IN      AAAA    ::1
ns      IN      A       192.168.1.10
```

Adjust the serial
Add ns and server A records pointing to server IP

a. **Domain name: *yourcomputername*.com or example.com**
b. **Server IP address: use ifconfig command to find this address**
c. **Server hostname: ns. *yourcomputername*.com or example.com**
d. **Webserver IP: same as Server IP**

4. **Test your DNS server**
    i. **Your server IP address xx.yy.zz.aa test your DNS configuration:**
        - **host IP address –** this should show your server IP backwards and the name
        - **dig IP address –** when run two times, you should see the second time a 0 second lookup time
        - **nslookup IP address –** you should see the server being used it the localhost (127.0.0.X) and it should resolve the name
        - **test all these with known website such as google.com or its IP address**

Note: host, dig, and nslookup do not show results as it is expected for DNS server because configured FQDN (www.example.com) is not online and just used for testing.

5. **The following link provides a step by step DNS configuration:**
   https://ubuntu.com/server/docs/service-domain-name-service-dns