

---

# Test Lab Guide

## Fedora Base Lab Configuration

### 1. Abstract

This Test Lab Guide (TLG) will provide the user with step-by-step instructions on how to configure a basic network configuration using Fedora servers and a Fedora client. Those acquainted with the Microsoft TLG documentation will find the format familiar. The resulting test lab environment will provide a stable base for building other test labs. It is recommended that users preserve the start of their test machines once this walk-through has been successfully completed. In a physical environment the hard drive of the machine can be imaged. In a virtual environment, the “snapshot” or equivalent feature can be used to preserve the current state of the operating system. Preserving the working conditions of the basic test lab will allow the user to experiment without fear of having to completely rebuild their environment. The ultimate goal of the TLG is to familiarize the user with the Fedora Operating system and how to deploy common network services based on that operating system, and ultimately enable the user to conduct their own experiments in a Fedora based environment.

### 2. Introduction

The purpose of the Test Lab Guides (TLGs) is providing users with practical guidelines for deploying current operating systems in a way that results in a functional configuration. Using a TLG will instruct the user in which servers to create, how to configure the operating systems and services, and how to install and configure additional software. A TLG experience enables the user to experience the entire set-up process from start to finish.

This TLG is written with the goals of reusability and extensibility in mind. The purpose of this particular TLG is to enable the creation of a basic network utilizing Fedora as the central operating system. Once this network is complete several other TLGs can be built on top of this base configuration.

Once this lab is completed, it would be wise to save the initial configuration. How this is best done will depend on how the test environment was originally deployed. A physical environment can be preserved by imaging the drives of each machine

to be stored in a separate location and retrieved when needed. In a network deployed virtually, a snapshot can be taken of each machine. This will preserve the current settings and configurations. Preserving the lab in a functional state is important, because it allows for a functional configuration to be restored without completely repeating the base lab configuration steps. This is helpful for correcting after a mistake or generating a test environment for a new product.

### 3. Disclaimer

This website contains work created for informational purposes. Information may be out of date, or changed or updated without notice. By using this website, you recognize and agree that all information is provided “AS IS” without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

The entire risk arising out of the use of these guides remains with you. IN NO EVENT SHALL CHAMPLAIN COLLEGE, ITS STUDENTS, FACULTY, OR ANYONE ELSE INVOLVED IN THE CREATION, PRODUCTION, OR DELIVERY OF THE GUIDES BE LIABLE TO ANY PERSON FOR ANY DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES FOR ANY USE OF THE MATERIAL ON THIS WEBSITE. This includes, without limitation, damages for lost profits, business interruption, loss of data or business information, damage to computer equipment or networks, or other loss arising out of the use of any information in the guides

### 4. Contents of This Guide

This document contains instructions for settings up the Fedora base configuration test lab by deploying two servers running Fedora, one client running Fedora and one firewall built on Fedora. Once completed, the lab will simulate the functioning of a private intranet.

**Important:** The instructions provided in this document are for educational purposes. They do not represent best practices nor are they recommendations for a production network. These configurations should not be put into place on a production network. This network should be deployed on a separate network specific to testing (physical or virtual).

## **4.1. Network Overview**

The Fedora Base Configuration test lab consists of the following:

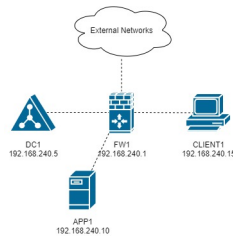
- One server running Fedora Server 27 named FW1
  - IP Address: 192.168.240.1
  - Gateway: N/A
  - DNS Server: 192.168.240.5
  - Hostname: firewall.business.com
  - Two NICs configured to handle traffic between the intranet and the external Internet connection
- One server running Fedora Server 27 named DC1
  - IP Address: 192.168.240.5
  - Gateway: 192.168.240.1
  - DNS Server: 192.168.240.5 (localhost)
  - Hostname: dc1.business.com
  - Configured as the intranet Domain Name System (DNS), and Dynamic Host Configuration Protocol (DHCP) server
- One server running Fedora Server 27 named APP1
  - IP Address: 192.168.240.10
  - Gateway: 192.168.240.1
  - DNS Server: 192.168.240.5
  - Hostname: app1.business.com
  - Configured on the intranet as a web and application server
- One client running Fedora Workstation 27 named CLIENT1
  - IP Address: 192.168.240.15
  - Gateway: 192.168.240.1
  - DNS Server: 192.168.240.5
  - Hostname: client1.business.com

- Will have ability to switch between the intranet and Internet subnets

The CentOS 7 Base Configuration TLG is comprised of one subnet:

- An intranet, known as the Internal subnet (192.168.240.0/24 in this example)

Network Diagram:



This document consists of four major parts:

- Step 1: Configure FW1
- Step 2: Configure DC1
- Step 3: Configure APP1
- Step 4: Configure CLIENT1

## **4.2. Hardware and Software Requirements**

The following are the minimum required components for deploying the test lab:

- An installer disc or .iso file for Fedora Server 27, 64 bit
- An installer disc or .iso file for Fedora Workstation 27, 64 bit
- One machine that will meet the minimum install requirements for Fedora Server with 2 NICs
- Two machines that will meet the minimum install requirements for Fedora Server
- One machine that will meet the minimum install requirements for Fedora Workstation

If the environment will be deployed in a virtualized environment, the virtual solution must support Fedora virtual machines (or virtual linux machines in general). The

server hardware must support the amount for RAM required to run the virtual operating systems included in the base configuration test lab with space for expansion as required by the additional TLGs.

## **5. Steps for Configuring the Network**

This test network consists of four machines:

1. FW1
2. DC1
3. APP1
4. CLIENT1

You must be logged on as a user who can execute sudo commands to complete this TLG. The steps to configure each machine are below.

### ***5.1. Step One - Configure FW1***

The FW1 machine will act as a firewall/router for the network. Configuring FW1 will consist of:

1. Installing the operating system - Fedora Server 27.
2. Configure TCP/IP properties.
3. Configure NAT rules.

### ***5.2. Installing the Operating System***

The first step is to install Fedora Server 27 on the machine that will be used as the router. The hardware for this machine must include two NICs if running the operating system in a physical environment. If utilizing a virtual environment, please ensure that the virtual machine for this system includes two network cards in its virtual hardware. Instructions on how to do this are below.

1. Start the installation using the installer disk or the .iso file and follow the installation prompts.
  - a. When using VMware to install the operating system, many of the option will be prefilled.

- i. VMware will prompt for a username and password - it will create a non-sudo user and assign the OS root password.
2. Before clicking the finish button on the final window, click “Customize Hardware...”
  - a. Click “Add...” at the bottom.
  - b. Click “Network Adapter” and then finish.
  - c. On “Network Adapter 2” click Custom.
  - d. Click close, and finish the installation.
3. During the installation process, create a user and give him administration rights. This is time sensitive, so make sure not to walk away from the OS while installing.
  - a. Click on “User Creation”.
  - b. Click on the box “Make this user administrator”.
  - c. Assign credentials.
  - d. Click Apply
4. Log on using the credentials created during the installation process

### **5.3. Configure TCP/IP Properties**

This operating has two NICs to be configured. One will act as the external network card and the other will act as the internal network card. It is important to be using an account that has sudo privileges or be in the root account, as many of the configurations require executive privilege. Commands noted with sudo can be executed without it if using a root user.

1. Ensure that Network Manager is running by running the command `systemctl status NetworkManager.service`. This command, like most commands, is caps sensitive.
  - a. The output should be Active: active (running) in green.
  - b. If not, enter the command `sudo systemctl start NetworkManager.service`
2. Install nmtui, if not already installed.

- a. Enter the command `sudo yum install networkManager-tui -y`.
3. Enter the command `sudo nmtui` to enter the Network Manager
  - a. Use the arrow keys to select "Edit Connection" for the internet-facing interface and then press "Enter"
    - i. Use the arrow keys to ensure that "IPv4 CONFIGURATION" is set to "<Automatic>"
    - ii. Use the arrow keys to ensure that "Automatically connect" has and "[x]" in the brackets next to it.
    - iii. Use the arrow keys to select "<Back>" and press Enter to return to the main menu.
  - b. Use the arrow keys to select "Edit Connection" for the internal-facing interface and press "Enter" to select.
    - i. Use the arrow keys to select "Manual" in the menu next to "IPv4 Configuration".
    - ii. Use the arrow keys to select "Show" next to the "IPv4 Configuration"
    - iii. Under "Addresses" add in 192.168.240.1/24 or an IP in the address range that you have selected for use in this lab.
    - iv. Use the arrow keys to ensure that "Automatically connect" has and "[x]" in the brackets next to it.
    - v. Use the arrow keys to select "<Back>" and press Enter to return to the main menu.
  - c. Use the arrow keys to select "Quit" and press Enter to exit the editor.
4. Set the hostname by running the command: `sudo hostnamectl set-hostname firewall.business.com`

## **5.4. Configure Routing rules**

Enable IP forwarding:

1. Enable ipv4 forwarding with the command: `sudo sysctl -w net.ipv4.ip_forward=1`

2. To make sure that this setting is enabled use the command `sudo sysctl net.ipv4.ip_forward`. The console should print out `net.ipv4.ip_forward = 1`.
  - a. If you restart your OS, you may have to reenter this command.
3. Recent editions of fedora use firewalld to manage networking; however, we are going to install and use iptables instead because it is a more direct communication with the networking functionality.
4. Install and enable iptables-services:
  - a. `sudo yum install iptables-services`
  - b. `sudo systemctl mask firewalld.service`
  - c. `sudo systemctl enable iptables.service`
5. Flush the current rules and NAT tables to ensure only the rules we create are being applied.
  - a. `sudo iptables -F`
  - b. `sudo iptables -t nat -F`
6. Configure the routing rules where eth0 is the external interface and eth1 is the internal interface.
  - a. `sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE`
  - b. `sudo iptables -A FORWARD -i eth0 -o eth1 -m state --state RELATED,ESTABLISHED -j ACCEPT`
  - c. `sudo iptables -A FORWARD -i eth1 -o eth0 -j ACCEPT`
7. Save the configuration using the command `sudo service iptables save`.



8. The configuration can be checked with the command `cat /etc/sysconfig/iptables`. The result will show only the commands entered in step six.

```
[root@router federal]# cat /etc/sysconfig/iptables
# Generated by iptables-save v1.6.1 on Sat Mar 3 14:07:53 2018
*nat
:PREROUTING ACCEPT [1:78]
:INPUT ACCEPT [1:78]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
-A POSTROUTING -o ens33 -j MASQUERADE
-A POSTROUTING -o ens33 -j MASQUERADE
COMMIT
# Completed on Sat Mar 3 14:07:53 2018
# Generated by iptables-save v1.6.1 on Sat Mar 3 14:07:53 2018
*filter
:INPUT ACCEPT [12:916]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [4:304]
-A FORWARD -i ens33 -o ens34 -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -i ens34 -o ens33 -j ACCEPT
COMMIT
# Completed on Sat Mar 3 14:07:53 2018
```

9. Stop `firewalld` and start `iptables`:

- a. `sudo systemctl stop firewalld.service`
- b. `sudo systemctl start iptables.service`

10. This configuration can be tested by completing the `client1` configuration in Step 2.

## 5.5. Step Two - Configure *CLIENT1*

This is typically the last step of a networking guide, but setting up all the services prior to having any means of testing can lead to a knot of configuration errors. By configuring the end user device initially, and adapting to the new network services continually, we can ensure that each service works as it goes up. Many of these settings will be changed in the future. A final configuration will be included in the appendix. Configuration steps for basic functionality include:

1. Install the operating system - Fedora Workstation 27
2. Configure the TCP/IP properties & Test Configuration

## 5.6. Install Operating System

1. Follow the prompts to install the operating system for Fedora Workstation 27.
  - a. Select the appropriate language and then select “Continue”
2. On the Installation Summary Page:

- a. Select the options most appropriate to your keyboard and time/date selection.



- b. Everything under “Software” can be left as defaults.
  - c. Under “System” select the “Installation Destination”. We will leave everything as the defaults, but most confirm the options by selecting “Done” in the top left corner.
  - d. Now the “Begin Installation” box in the bottom right should have turned blue. Click on this button.
3. While the operating system is installing create a root password and a user account.
    - a. This is done by selecting each option, filling out the required information and hitting the “Done” button in the top left corner. Please remember these accounts as they are how the machine will be accessed.
  4. At the end of the installation process you will be prompted to reboot the machine. After it reboots you will be able to login with the credentials setup in step three.
  5. On the first reboot you will be prompted to re-select some options. After going through this process the client will be ready to use.

## **5.7. Configure TCP/IP properties & Test Configuration**

1. The network settings will be configured through the graphical user interface.
2. Select the drop down menu in the top right corner by clicking on the small arrow next to the power symbol.
  - a. Select “Wired Connection” and click on “Wired Settings”
  - b. This will open a new dialogue box. Click on the small box with a gear under the “Wired” settings.

- c. This opens a new dialogue box. Select “Ipv4” along the top bar.
  - d. Next to “IPv4” options select “Manual”
  - e. Fill out the “Addresses” box with the IP address you wish to assign to the client, the netmask of the network (255.255.255.0) and the Gateway (the address of the router).
  - f. Set DNS to “8.8.8.8”
  - g. Select “Apply” in the top right corner to apply the settings
  - h. Back on the Network Settings dialog box, click on the “On” button to turn the connection off. It should turn from blue to grey. Select once again to turn the settings back on. This ensures the settings are fully applied.
3. If the network is configured properly and the router is functioning, the client will now be able to function.
- a. Open a terminal by selecting “Activities” in the top left corner and typing cmd in to the search box. When “terminal” appears select it. This will open a command prompt.
  - b. Enter the command `ping 8.8.8.8` at the command line to check network connectivity. This should generate continuous responses. Use `ctrl + c` to exit.
  - c. Enter the command `ping www.google.com` to test hostname resolution. This should generate continuous positive responses, and use `ctrl + c` to exit.
  - d. When both commands resolve properly it will be safe to preserve the state of the router using VMware’s snapshot function or similar application.

The final configuration of client1 is listed in the appendix. However, it is recommended that users follow along with the guide, making changes as necessary. Jumping ahead to the final configuration may result in failure when testing service functionality in the next few sections.

## **5.8. Step Three - Configure DC1**

The DC1 machine will act as a DNS server and a DHCP server. Configuring DC1 will be include:

1. Installing an operating system - Fedora Server 27.
2. Configure TCP/IP properties.
3. Install and Configure DNS.
4. Re-configure TCP/IP properties to account for new DNS server.
5. Install and Configure DHCP.
6. Configure DHCP and DNS settings on Client1.
7. Snapshot the Configuration.

## **5.9. Install the Operating System**

1. Start the installation using the installer disk or the .iso file and follow the installation prompts.
  - a. When using VMware to install the operating system, many of the option will be prefilled.
    - i. VMware will prompt for a username and password - remember these credentials!
2. Log on using the credentials created during the installation process

## **5.10. Configure TCP/IP**

1. Check that the Network Manager is running using the command `systemctl status NetworkManager.service`.
  - a. If the Network Manager is running the result will be `Active: active (running)` in green.
  - b. If the Network Manager is not running, enter the command `systemctl start NetworkManager.service`.
2. Install `nm-tui`, if not already installed.
  - a. Enter the command `sudo yum install NetworkManager-tui -y`.
    - i. You may have to temporarily connect directly to the internet to do this.
3. Enter the command `sudo nm-tui` to enter the Network Manager graphical user interface.

- a. Use the arrows keys to select "Edit Connection" for the Internet-facing interface and then hit Enter.
  - b. Use the arrow keys to set "IPv4 CONFIGURATION" and set it to "Manual"
  - c. Set "Addresses" with an IP address in your configured subnet that does not conflict with any previously assigned. Add the subnet in slash notation on the end, or the configuration will fail.
    - i. Example: 192.168.240.5/24
  - d. Set "DNS Server" to the address "8.8.8.8"
  - e. Set "Gateway" to the address of the router
  - f. Ensure that "Automatically connect" has an "[x]" in the brackets next to it.
  - g. Use the arrow keys to select "OK" and return to the main menu of the editor.
  - h. Use the arrow keys to select "Quit" and press Enter to exit the editor
4. Edit the hostname using the command `sudo hostnamectl set-hostname dc1.business.com`
  5. Ensure that the connection works with the commands `ping 8.8.8.8` and `ping www.google.com`. Use `ctrl + c` to stop the command execution. Both should produce results if the network is properly configured.
    - a. You may have to restart the network to do this.
      - i. This can be done by entering the command `sudo systemctl restart Network`.

These are temporary settings that will ensure internet connectivity while downloading the required packages for DNS and DHCP. The settings will be reconfigured after the appropriate services have been set up.

## **5.11. Install/Configure DNS**

1. Before we do anything, open the DNS port of 53/tcp
  - a. Enter command `sudo firewall-cmd --open-port=53/tcp --permanent`.
  - b. Enter command `sudo firewall-cmd --reload`.

2. Install bind by typing the command `sudo yum install bind bind-utils -y` and hitting enter.
3. Configure `etc/named.conf` (this is the configuration file for BIND)
  - a. Under options {
    - i. Edit `listen-on port 53` to add in the IP address of the DNS server
      - A. It should look like `listen-on port 53 { 127.0.0.1; your-dc1-address; };`
    - ii. Comment out `listen-on-v6 port 53 { ::1; };` by adding a # in front
      - A. It should look like this `#listen-on-v6 port 53 { ::1; };`
    - iii. Edit `allow-query` to add in the subnet
      - A. It should look like `allow-query { localhost; 192.168.240.0/24; };`
    - iv. Add in forwarders with IP addresses of 8.8.8.8 and 8.8.4.4
      - A. It should look like `forwarders { 8.8.8.8; 8.8.4.4; };`
    - v. At the end of the file add in `include "/etc/named/named.conf.local";`
4. Save this file by using `ctrl + x` and entering `y` at the prompt.
5. Create the `named.conf.local` file and add:

```
[root@localhost federal]# cat /etc/named/named.conf.local
zone "business.com" {
    type master;
    file "/var/named/db.business.com";
    allow-query { any; };
};

zone "240.168.192.in-addr.arpa" {
    type master;
    file "/var/named/db.240.168.192";
    allow-query { any; };
};
```

- a. Make sure the second one matches your subnet's addressing scheme

6. Create the two files referenced in named.conf.local

a. Enter commands:

- i. `sudo touch /var/named/db.business.com`
- ii. `sudo touch /var/named/db.240.168.192`

7. Make a copy of a default forwarder named. file to use as a template by running the command `sudo cp /var/named/named.empty <file path here>`

- a. Next edit this file by running the command `sudo nano /etc/named/db.business.com`. This will bring you into the Nano editor where you will want to make changes to what is already there.
- b. Change the `$TTL` value of `3h` to `604800` then replace `name.invalid.` with `dc1.business.com.` `YourUser.business.com.` Remove the `@` symbol before this value.
- c. Next change the values for serial to the days date with a serial number at the end (ie. `3/3/17 = 20170303+ serial #`). The serial number can be any value. I used `02`.
- d. Now changes the values for refresh, retry, expire, and minimum to `604800`, `86400`, `2419200`, and `604800` respectively
- e. Add in `@` followed by a tab then `IN` with another tab for the `NS`, `A`, and `AAAA` lines also change the `@` in the `NS` line to `dc1.business.com.`
- f. Add the `A` record addresses as shown below:

```
firewall IN A 192.168.240.1

dc1      IN A 192.168.240.5

app1     IN A 192.168.240.10

client1  IN A 192.168.240.15
```

g. Add in CNAME Record addresses as shown below:

```
server1 IN CNAME dc1.business.com.
```

```
server2 IN CNAME app1.business.com.
```

h. It should look like the file below:

```
$TTL 604800
@      IN      SOA      dc1.business.com. champlain.business.com. (
                                2020040702; serial
                                604800 ; refresh
                                186400 ; retry
                                2419200 ; expire
                                604800 ); minimum

@      NS      dc1.business.com.
@      A       127.0.0.1
@      AAAA    ::1

firewall      IN      A       192.168.240.1
dc1            IN      A       192.168.240.5
app1           IN      A       192.168.240.10
client1        IN      A       192.168.240.15

server1        IN      CNAME   dc1.business.com.
server2        IN      CNAME   app1.business.com.
```

i. Press `ctrl + x` to exit the editor and then hit `y` to save the file

8. Repeat the steps and create this file:

```
$TTL 604800
@      IN      SOA      dc1.business.com. champlain.business.com. (
                                2020040702; serial
                                604800 ; refresh
                                86400  ; retry
                                2419200 ; expire
                                604800 ); minimum

@      IN      NS      dc1.business.com.
@      IN      A       192.168.240.5
1      IN      PTR      firewall.business.com.
2      IN      PTR      dc1.business.com.
3      IN      PTR      app1.business.com.
4      IN      PTR      client1.business.com.
```

9. Now it is time to start the DNS server

- a. Enter the command `sudo systemctl enable named` to enable the DNS server.
- b. Then enter the command `sudo systemctl start named` to start the DNS server



## 5.12. Re-configure TCP/IP

After DNS has been configured, the TCP/IP settings of DC1 can be reconfigured to point to itself as the DNS server rather than the router.

1. Install nmtui, if not already installed.
  - a. Enter the command `sudo yum install NetworkManager-tui -y`.
    - i. You may have to temporarily connect directly to the internet to do this.
2. Enter the command `sudo nmtui` to enter the Network Manager interface.
  - a. Use the arrow keys to select "Edit Connection" for the interface and press the enter key
  - b. Use the arrow keys to select "IPv4 Configuration" and hit Enter
    - i. Under "DNS servers" replace the router address with 127.0.0.1
  - c. Use the arrow keys to select "<OK>" and press Enter
  - d. Use the arrow keys to select "<Back>" and press Enter to return to the main menu of the editor
  - e. Use the arrow keys to select "quit" and press Enter to exit the editor.
3. Restart the network by running the command `sudo systemctl restart network`.
4. Ensure that the connection works with the commands `ping 8.8.8.8` and `ping www.google.com`. Both should produce results if the network is properly configured and can be exited with `ctrl + c`.

## 5.13. Install and Configure DHCP

1. Install DHCP using the command `sudo yum install dhcp -y`
2. Enter the configuration file by entering the command `sudo nano /etc/dhcp/dhcpd.conf`.
  - a. Enter the following configuration lines:
    - i. Please note: ensure that you are using addresses that match your network configuration.

- ii. Please note II: the hardware ethernet when making reservations for client1 is the MAC address of client1. This can be found by running the `ifconfig` command on client1

```
# Create new domain
option domain-name "business.com";

# Specify DNS server IP
option domain-name-servers 192.168.240.5;

# Specify default lease time
default-lease-time 600;

# Specify max lease time
max-lease-time 7200;

# Specify router
option routers 192.168.240.1;

# Specify broadcast address
option broadcast-address 192.168.240.255;

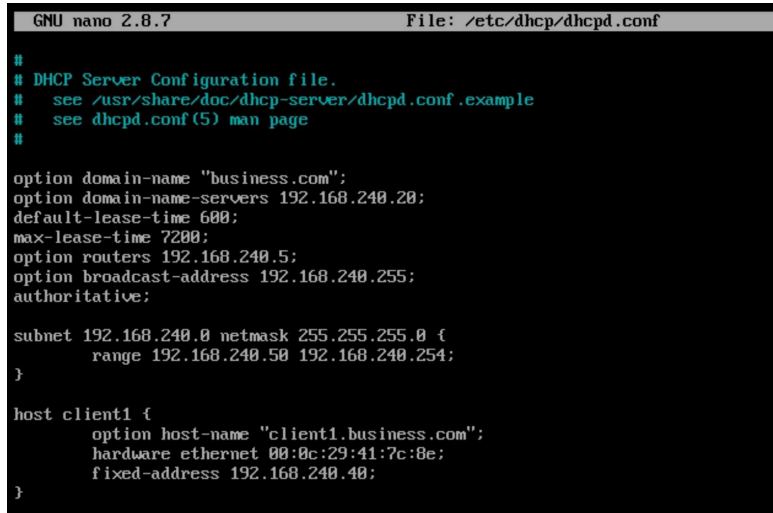
# Declare this as authoritative
authoritative;

# Specify network range
subnet 192.168.240.0 netmask 255.255.255.0 {
    range 192.168.240.15 192.168.240.254;
}

# Make IP Address reservation for client1
host client1 {
    option host-name "client1.business.com";
    hardware Ethernet 00:50:56:AF:9C:E3; (Your client1 mac
    address)
    Fixed-address 192.168.240.15;
```

```
}
```

An example is shown below:



```
GNU nano 2.8.7                               File: /etc/dhcp/dhcpd.conf

#
# DHCP Server Configuration file.
#   see /usr/share/doc/dhcp-server/dhcpd.conf.example
#   see dhcpd.conf(5) man page
#

option domain-name "business.com";
option domain-name-servers 192.168.240.20;
default-lease-time 600;
max-lease-time 7200;
option routers 192.168.240.5;
option broadcast-address 192.168.240.255;
authoritative;

subnet 192.168.240.0 netmask 255.255.255.0 {
    range 192.168.240.50 192.168.240.254;
}

host client1 {
    option host-name "client1.business.com";
    hardware ethernet 00:0c:29:41:7c:8e;
    fixed-address 192.168.240.40;
}
```

- b. Press `ctrl + x` to exit the Editor and then enter `y` to save the file.
3. Start the DHCP server using the following commands:
  - a. `sudo systemctl enable dhcpd` to enable the DHCP server
  - b. `sudo systemctl start dhcpd` to start the DHCP server At this point the DHCP server can be tested using the Client1 machine to ensure that everything functions properly.

## 5.14. Configure DHCP and DNS settings on Client1

The network settings will be configured through the graphical user interface on the client machine set up in Step 2.

1. Select the drop down menu in the top right corner by clicking on the small arrow next to the power symbol.
  - a. Select "Wired Connection" and click on "Wired Settings"
  - b. This will open a new dialogue box. Click on the small box with a gear under the "Wired" settings.
  - c. This opens a new dialogue box. Select "Ipv4" along the top bar.

- d. Next to “IPv4” options select “DHCP” and enter the address of AD1
  - e. Set DNS to the address of AD1.
  - f. Select “Apply” in the top right corner to apply the settings
  - g. Back on the Network Settings dialog box, click on the “On” button to turn the connection off. It should turn from blue to grey. Select once again to turn the settings back on. This ensures the settings are fully applied.
2. If the network is configured properly and the router is functioning, the client will now be able to function.
- a. Open a terminal by selecting “Activities” in the top left corner and typing `cmd` into the search box. When “terminal” appears select it. This will open a command prompt.
  - b. Enter the command `ping 8.8.8.8` at the command line to check network connectivity. This should generate continuous responses. Use `ctrl + c` to exit.
  - c. Enter the command `ping www.google.com` to test hostname resolution. This should generate continuous positive responses, and use `ctrl + c` to exit.
  - d. When both commands resolve properly it will be safe to preserve the state of the router using VMware’s snapshot function or similar application.

### ***5.15. Snapshot the Configuration***

Preserving a functional state of the machine is important in case changes are made accidentally. This can be done using VMWare’s snapshot functionality or other software packages for hardware labs.

### ***5.16. Step Four - Configure APP1***

APP1 will provide a web server and a Samba Share. Configuration steps for APP1 include:

1. Install the operating system - Fedora Server 27.
2. Configure TCP/IP Properties
3. Configure the web server

4. Configure the Samba Share
5. Test Access from client1
6. Snapshot the Configuration

### **5.17. Install the Operating System**

1. Start the installation using the installer disk or the .iso file and follow the installation prompts.
  - a. When using VMware to install the operating system, many of the option will be prefilled.
    - i. VMware will prompt for a username and password - remember these credentials!
  - b. Log on using the credentials created during the installation process

### **5.18. Configure TCP/IP Properties**

1. Check that the Network Manager is running using the command `systemctl status NetworkManager.service`.
  - a. If the Network Manager is running the result will be `Active: active (running)` in green.
  - b. If the Network Manager is not running, enter the command `systemctl start NetworkManager.service`
2. Install `nmtui`, if not already installed.
  - a. Enter the command `sudo yum install NetworkManager-tui -y`.
    - i. You may have to temporarily connect directly to the internet to do this.
3. Enter the command `sudo nmtui` to enter the Network Manager graphical user interface.
  - a. Use the arrows keys to select "Edit Connection" for the Internet-facing interface and then hit Enter.
  - b. Use the arrow keys to set "IPv4 CONFIGURATION" and set it to "Manual"

- c. Set "Addresses" with an IP address in your configured subnet that does not conflict with any previously assigned. Add the subnet in slash notation on the end, or the configuration will fail.
    - i. Example: 192.168.240.10/24
  - d. Set "DNS Server" to the address of the server AD1
  - e. Set "Gateway" to the address of the router
  - f. Ensure that "Automatically Connect" has an "[x]" in the brackets next to it.
  - g. Use the arrow keys to select "OK" and return to the main menu of the editor. ..Use the arrow keys to select "Quit" and press Enter to exit the editor
4. Edit the hostname using the command `sudo hostnamectl set-hostname app1.bussiness.com`
  5. Ensure that the connection works with the commands `ping 8.8.8.8` and `ping www.google.com`. Both should produce results if the network is properly configured.

### ***5.19. Configure the web server***

1. Run the command `sudo yum install httpd`
  - a. Enter y when prompted to confirm install
2. Run the command `sudo systemctl enable httpd`
3. Run the command `sudo systemctl start httpd`
4. Ensure that HTTP traffic will be allowed by allowing it through the firewall
  - a. Enter the command `sudo firewall-cmd --permanent --add-service=http`
  - b. Enter the command `sudo firewall-cmd --reload`

### ***5.20. Configure the samba share***

1. Start by installing Samba by using the following command `sudo yum install samba samba-client samba-common`
  - a. Enter y when prompted to confirm install.

2. Create a backup of the default configuration by running the following command  
`sudo mv /etc/samba/smb.conf /etc/samba/smb.conf.bak`
3. Create a new configuration file by entering the command `sudo nano /etc/samba/smb.conf`
  - a. Create an open (not secured share) by adding the following into the file, which will create definitions for a share that is accessible by all users on the LAN located at `/samba/allaccess`:

```
[global]
workgroup = WORKGROUP
server string = Samba Server %v
netbios name = app1
security = user
map to guest = bad user
dns proxy = no
hosts allow = 192.168.240.0/24
protocol = SMB3

[allaccess]
path = /samba/allaccess
browsable = yes
writable = yes
guest ok = yes
```

```
read only = no
```

- b. Press `ctrl + x` to exit the Editor and then hit `y` to save the file.
- c. The file should look like the one shown below:



```
[global]
workgroup = WORKGROUP
server string = Samba Server %v
netbios name = app1
security = user
map to guest = bad user
dns proxy = no
hosts allow = 192.168.240.0/24

[allaccess]
path = /samba/allaccess
browsable = yes
writable = yes
guest ok = yes
read only = no
```

4. Now create the share directory by running the command `sudo mkdir -p /samba/allaccess`
5. Add SMB permissions to a user in your system.
  - a. Enter command `sudo smbpassword -a <user>`
6. You will also need to allow Samba traffic through the firewall, use the following commands to do so:
  - a. `sudo firewall-cmd --permanent --add-service=samba`
  - b. `sudo firewall-cmd --reload`
7. Now you need to edit the permissions of the share so guest users can access it, start by moving to the `/samba` directory by using the `cd /samba` command
  - a. Next run the command `sudo chmod -R 0755 allaccess/`
  - b. Then run `sudo chown -R nobody:nobody allaccess/`



8. Now you need to configure SELinux to allow access to the Samba directory, use the following command to do so `sudo chcon -t samba_share_t allaccess/`
9. Start and enable Samba services using the following commands:
  - a. `sudo systemctl enable smb`
  - b. `sudo systemctl enable nmb`
  - c. `sudo systemctl start smb`
  - d. `sudo systemctl start nmb`

This finishes the creation of the Samba share that is accessible to all users on the LAN.

## 5.21. Test web server access

1. Log on to the client machine and open the mozilla web browser.
2. Enter the IP address of the app1 machine to the address bar.
3. This should present the default Apache homepage as shown below.



## 5.22. Test samba share access

1. Install on client1 `sudo yum install samba-client samba-common cifs-utils` using the command line.
2. Have to let through the firewall: `sudo firewall-cmd --permanent --dd-service=samba` and `firewall-cmd --reload`

```
session setup failed: WERR_ACCESS_DENIED
[fedora@client1 ~]$ smbclient //192.168.240.30/allaccess -U user1
Enter SAMBA\user1's password:
Try "help" to get a list of possible commands.
smb: \>
```

## 5.23. Snapshot the Configuration

Preserving a functional state of the machine is important in case changes are made accidentally. This can be done using VMWare's snapshot functionality or other software packages for hardware labs.

## 6. Additional Resources

### 6.1. How to give a user sudo privileges

Most of the steps taken in this base lab require using the command `sudo`, which can only be executed by a user with those privileges.

1. Log on to a root account.
2. Execute the command `usermod username -a -G wheel` where `username` is the name of the user you'd like to elevate the privileges of.
3. Log out of the root account using the command `exit`.
4. Log in to the account that was granted sudo privileges.
5. Test access using the command `sudo -v`
  - a. This will prompt a message and ask for the user's password if the process worked. Otherwise it will display an error message.

### 6.2. Final Client Configuration

Included in this section is information regarding the final network configuration of the client workstation. These settings can be accessed by logging in to the client workstation and selecting drop down menu in the top right corner and then selecting "Wired Settings" under the Wired Connected section. This will open a new dialogue box. To access settings select the box with the gear icon under the Wired section.

Below is an example of what the “Details” tab. The default route is the address of the router, and the DNS points to the address of DC1

The screenshot shows the 'Wired' network settings window with the 'Details' tab selected. The window has a 'Cancel' button on the top left and an 'Apply' button on the top right. The 'Details' tab is active, showing the following settings:

- Link speed: 1000 Mb/s
- IPv4 Address: 192.168.240.40
- IPv6 Address: fe80::f6cb:2fe5:3788:a515
- Hardware Address: 00:0C:29:41:7C:8E
- Default Route: 192.168.240.5
- DNS: 192.168.240.20

At the bottom, there are two checked checkboxes: 'Connect automatically' and 'Make available to other users'. A red button labeled 'Remove Connection Profile' is located at the bottom right.

The IPv4 tab settings are shown below. The address is assigned automatically by the DHCP server and the DNS settings show on the above screen are assigned here.

The screenshot shows the 'Wired' network settings window with the 'IPv4' tab selected. The window has a 'Cancel' button on the top left and an 'Apply' button on the top right. The 'IPv4' tab is active, showing the following settings:

- IPv4 Method**: Radio buttons for 'Automatic (DHCP)' (selected), 'Link-Local Only', 'Manual', and 'Disable'.
- DNS**: A dropdown menu set to 'Automatic' with an 'ON' button next to it.
- DNS Address**: A text box containing '192.168.240.20'.

Below the text box, there is a small note: 'Separate IP addresses with commas'.

These settings will enable the client workstation to be fully functional in the context of this network environment.

**Contributors.**

Evan Callaghan Michael Pare