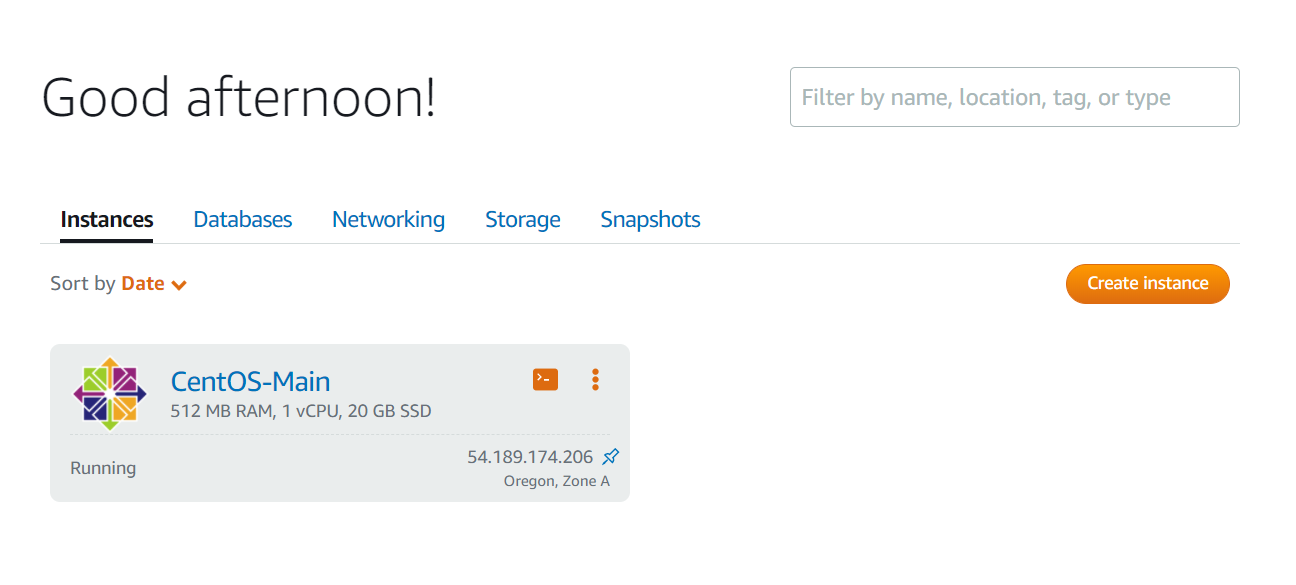
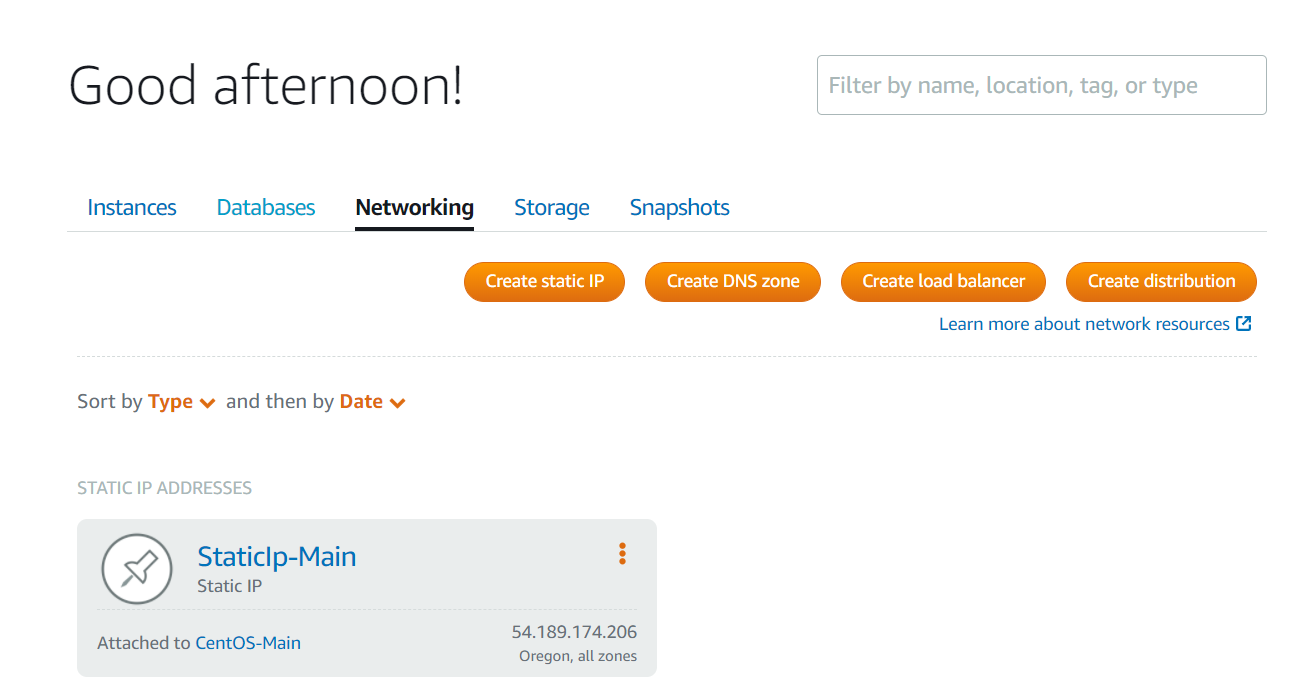
**Linux Project\_Group 6**

(Mohsen Zirrahi, Sohail Umer, Sophie Tan)

**1. Create a Virtual Machine as per these guidelines:**

* Free tier compliant VM specifications (CPU / RAM / Storage)
* Linux OS only deployment, no pre-installed software
  + Your choice of distribution
* Attach a publicly available (static if possible) IP address





**2. Install the LEMP stack (Nginx, PHP, MySQL or MariaDB) on the Virtual Machine.**

## Step 1 — Installing Nginx

To add the CentOS 7 EPEL repository, run the following command:

sudo yum install epel-release

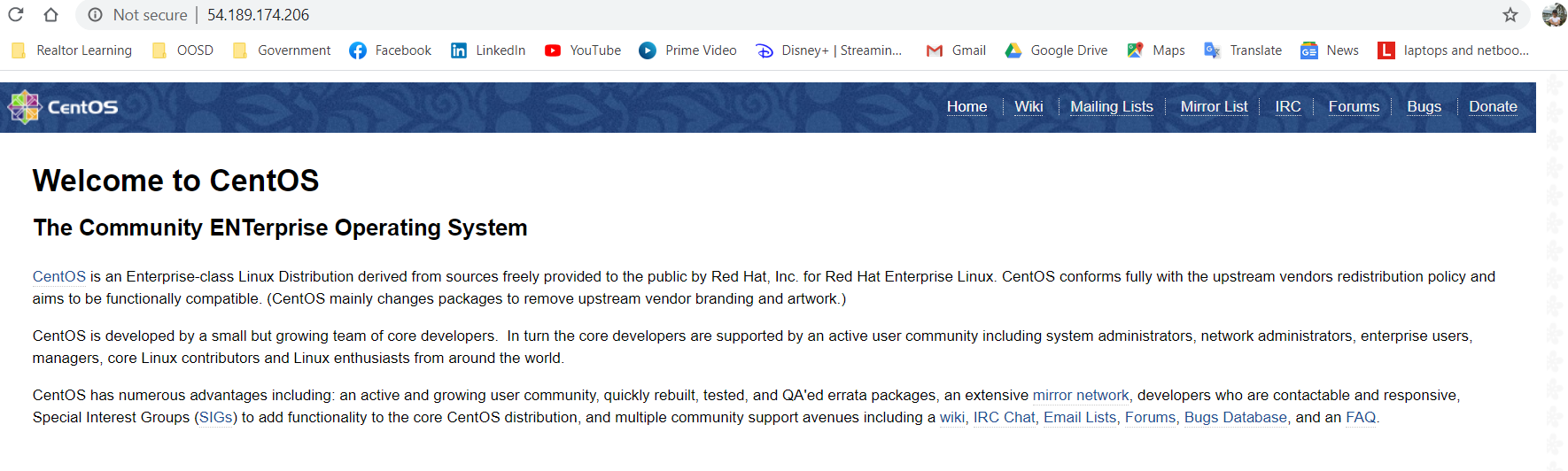
Now that the EPEL repository is installed on your server, install Nginx using the following yum command:

sudo yum install nginx

Once the installation is finished, start the Nginx service with:

sudo systemctl start nginx

Visit the server’s public IP address in web browser



To enable Nginx to start on boot, run the following command:

sudo systemctl enable nginx

## Step 2 — Installing MariaDB

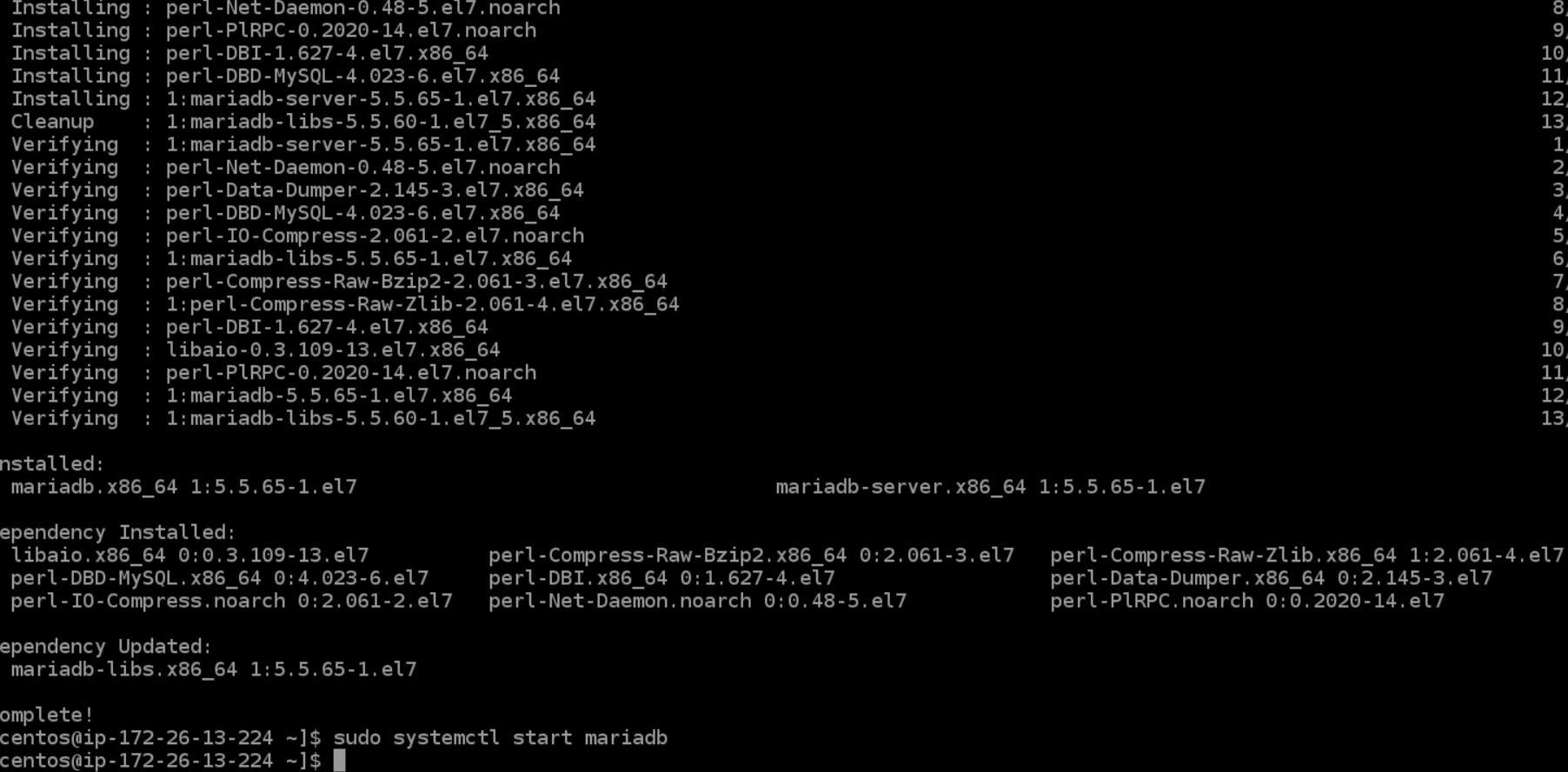
MariaDB is a community-developed fork of the MySQL relational database management system.

Install some other helper packages that will assist us in getting components to communicate with each other:

sudo yum install mariadb-server mariadb

Start MariaDB with the following command:

sudo systemctl start mariadb



## Step 3 — Installing PHP

PHP is the component of our setup that will process code to display dynamic content. It can run scripts, connect to our MySQL databases to get information, and hand the processed content over to our web server to display.

[Remi](https://rpms.remirepo.net/) is a popular package repository providing the most up-to-date PHP releases for CentOS servers.To install the Remi repository for CentOS 7, run:

sudo yum install <http://rpms.remirepo.net/enterprise/remi-release-7.rpm>

Run a command to enable the repository containing preferred version of PHP. To check which PHP 7+ releases are available in the Remi repository, run:

yum --disablerepo="\*" --enablerepo="remi-safe" list php[7-9][0-9].x86\_64

To enable the correct Remi package to get PHP 7.4 installed, run:

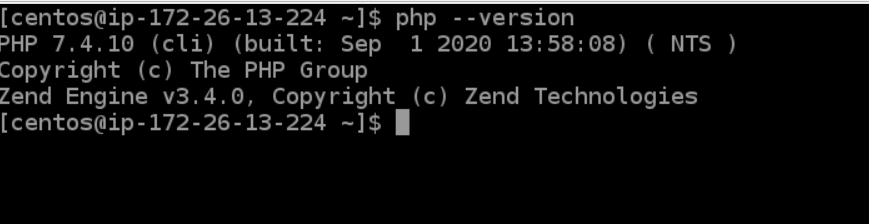
sudo yum-config-manager --enable remi-php74

The following command will install all the required packages to get PHP 7.4 set up within Nginx and allow it to connect to MySQL-based databases:

sudo yum install php php-mysqlnd php-fpm

To confirm that PHP is available as your chosen version, run:

php –version



To facilitate editing files on CentOS, first install nano, a more user-friendly text editor than vi:

sudo yum install nano

Open the /etc/php-fpm.d/www.conf configuration file using nano：

sudo nano /etc/php-fpm.d/www.conf

To enable and start the php-fpm service, run:

sudo systemctl start php-fpm

## Step 4 — Configuring Nginx to Process PHP Pages

First, open a new file in the /etc/nginx/conf.d directory:

sudo nano /etc/nginx/conf.d/default.conf

Copy the following PHP server definition block to your configuration file, and don’t forget to replace the server\_name directive so that it points to your server’s domain name or IP address:

server {

listen 80;

listen 443；

server\_name http://34.219.119.67/;

root /usr/share/nginx/html;

index index.php index.html index.htm;

location / {

try\_files $uri $uri/ =404;

}

error\_page 404 /404.html;

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root /usr/share/nginx/html;

}

location ~ \.php$ {

try\_files $uri =404;

fastcgi\_pass unix:/var/run/php-fpm/php-fpm.sock;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

}

 Restart Nginx to apply the changes:

sudo systemctl restart nginx

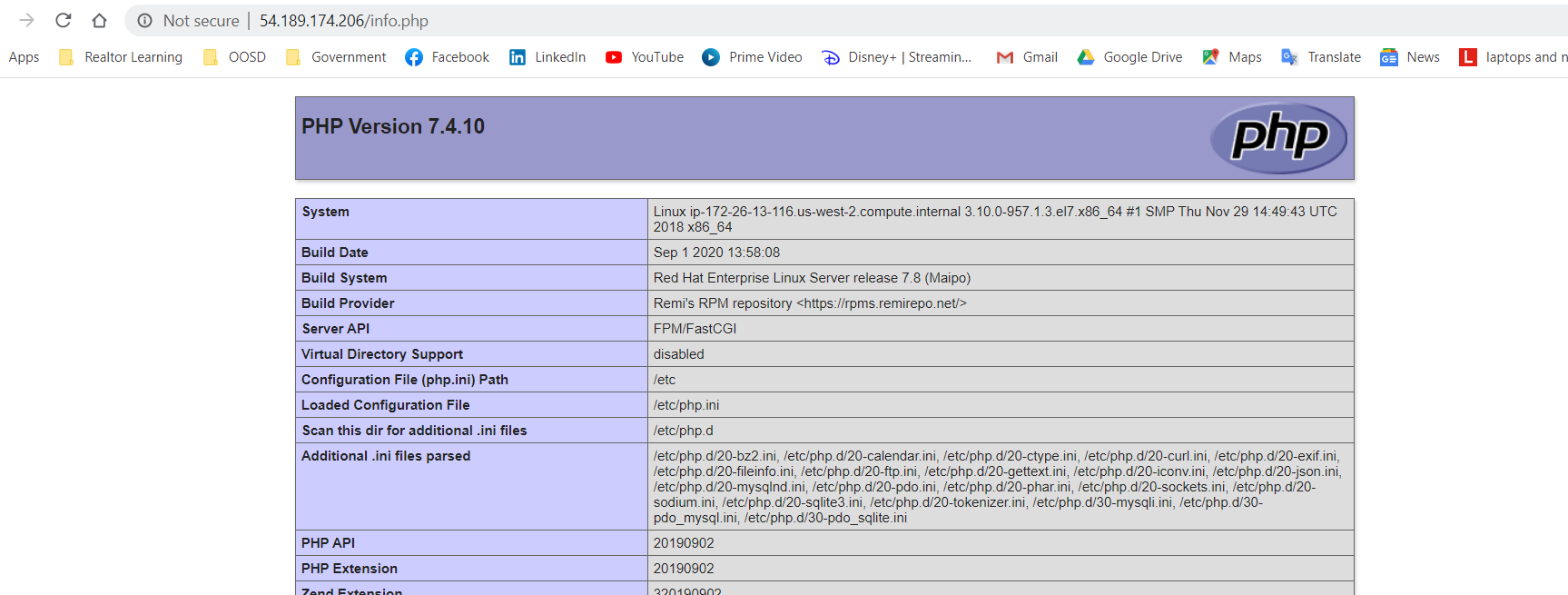
## Step 5 — Testing PHP Processing on the Web Server

Before creating script, make a change to the default ownership settings on Nginx’s document root, so that regular sudo user is able to create files in that location.

sudo chown -R centos.centos /usr/share/nginx/html/

Create a new PHP file called info.php at the /usr/share/nginx/html directory:

nano /usr/share/nginx/html/info.php



## Step 6 — Install the firewall

By default, firewalld is included in the “core” rpm group, but if in case it is not installed, you can always install it using yum.

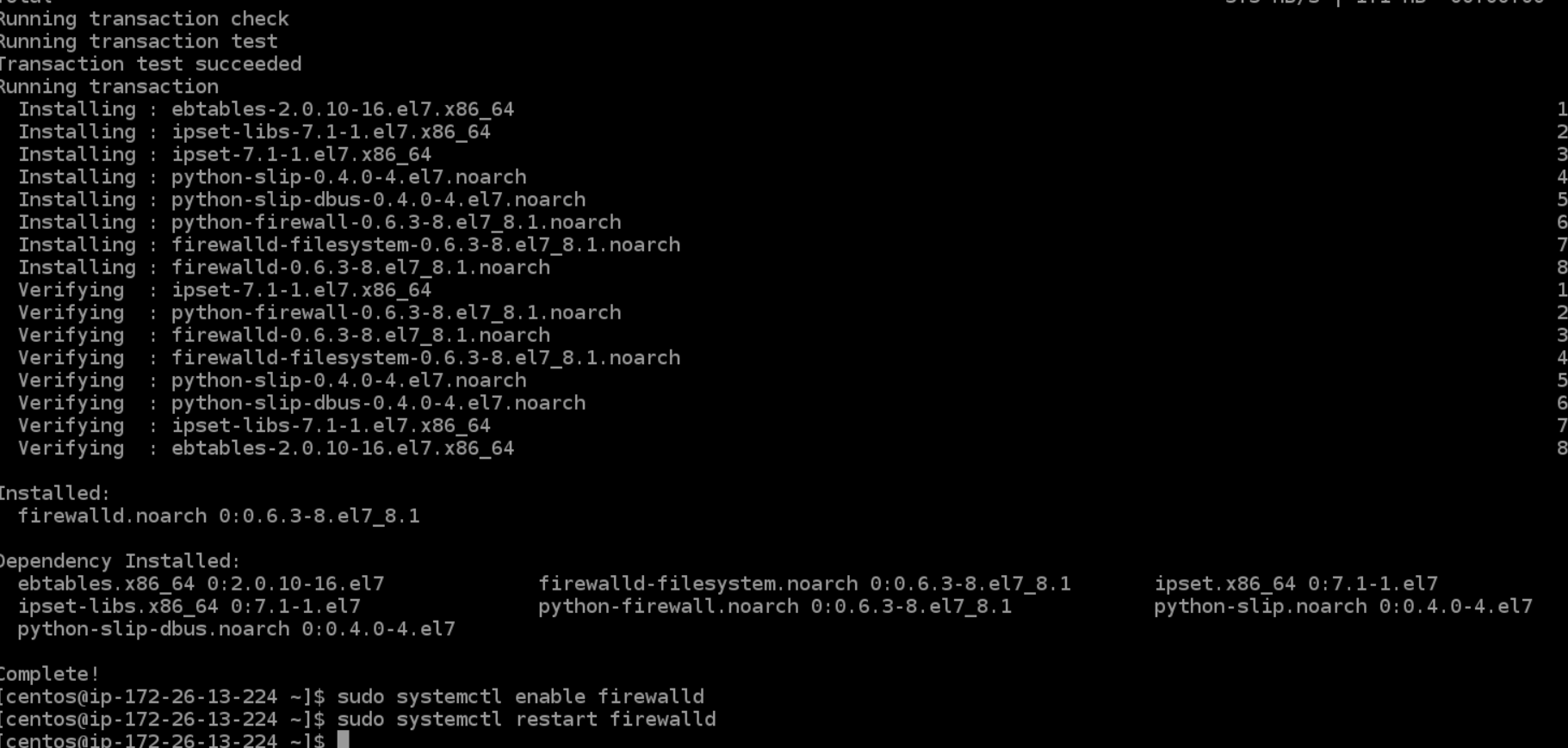
Sudo yum install -y firewalld

Enable the firewalld to start at boot:

Sudo systemctl enable firewalld

Restart the firewalld service now:

Sudo systemctl restart firewalld

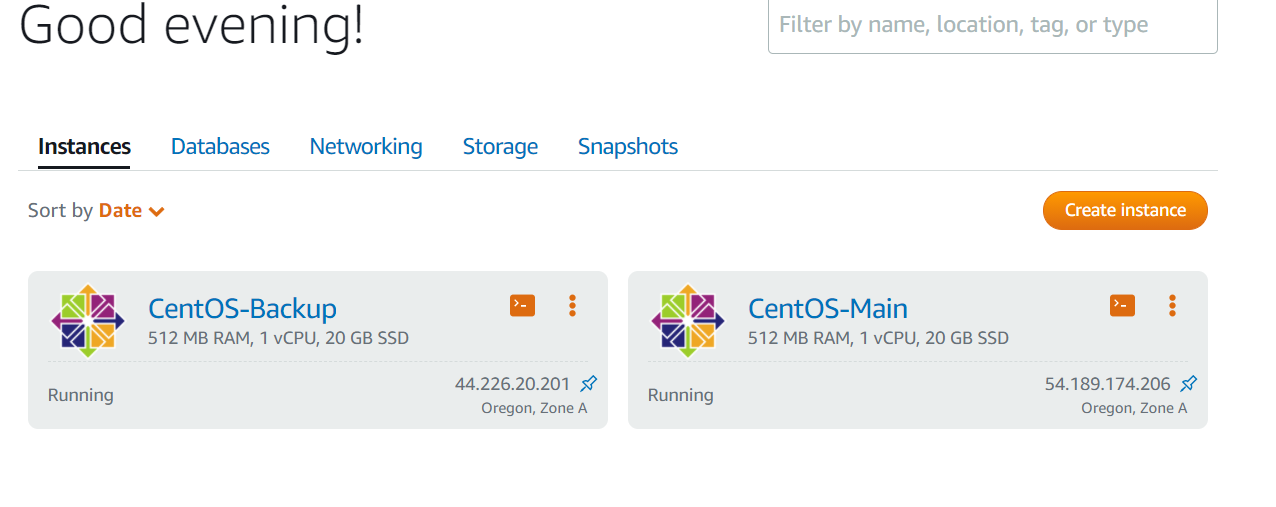


**2. Create a Virtual Machine as per these guidelines:**

* Free tier compliant VM specifications (CPU / RAM / Storage), or locally hosted on

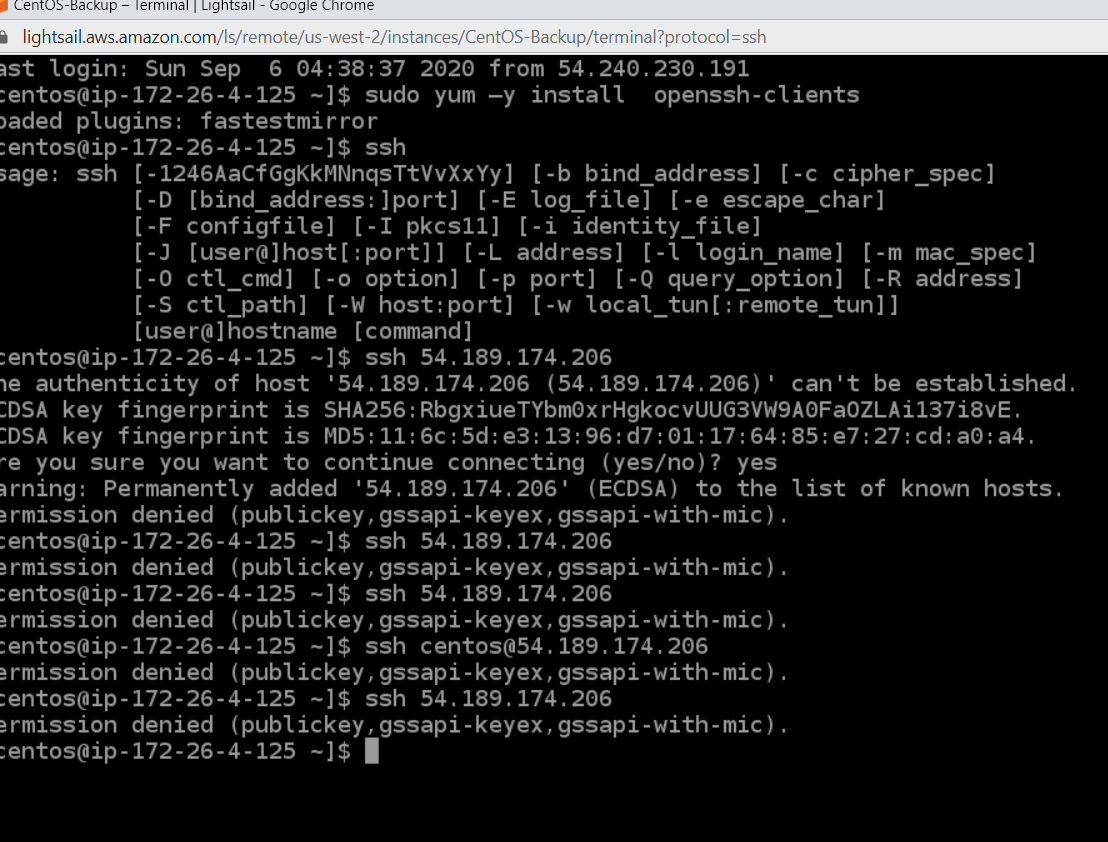
your laptop.

* Linux OS only deployment, no pre-installed software
* Your choice of distribution
* Attach a static IP address or be aware of the public IP address and network
* mechanism to allow connections to the web server.

****

**Hi Jay,**

**We are stuck with the ssh connection between the SSH server and client here. I will fix it up these two days and hand it to you by email. Now I don’t have time now. It’s 11:18PM. Thank you!**

****