NOPCOMMERCE APPLICATION

(docs.nopcommerce.com/en/installation-and-upgrading/installing-nopcommerce/installing-on-linux.html#installing-on-linux)

Start an instance on aws .

Connect to the linux machine and run the

**Sudo apt update** (system tracks changes to each package when a new version is released.)



to run nopcommerce application we need .net. So install .net with below command

open the terminal and run the following commands

wget https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

Update the products available for installation, then install the .NET runtime:

sudo apt-get update

sudo apt-get install -y apt-transport-https aspnetcore-runtime-7.0

You can see all installed .Net

dotnet --list-runtimes

Install the nginx package:

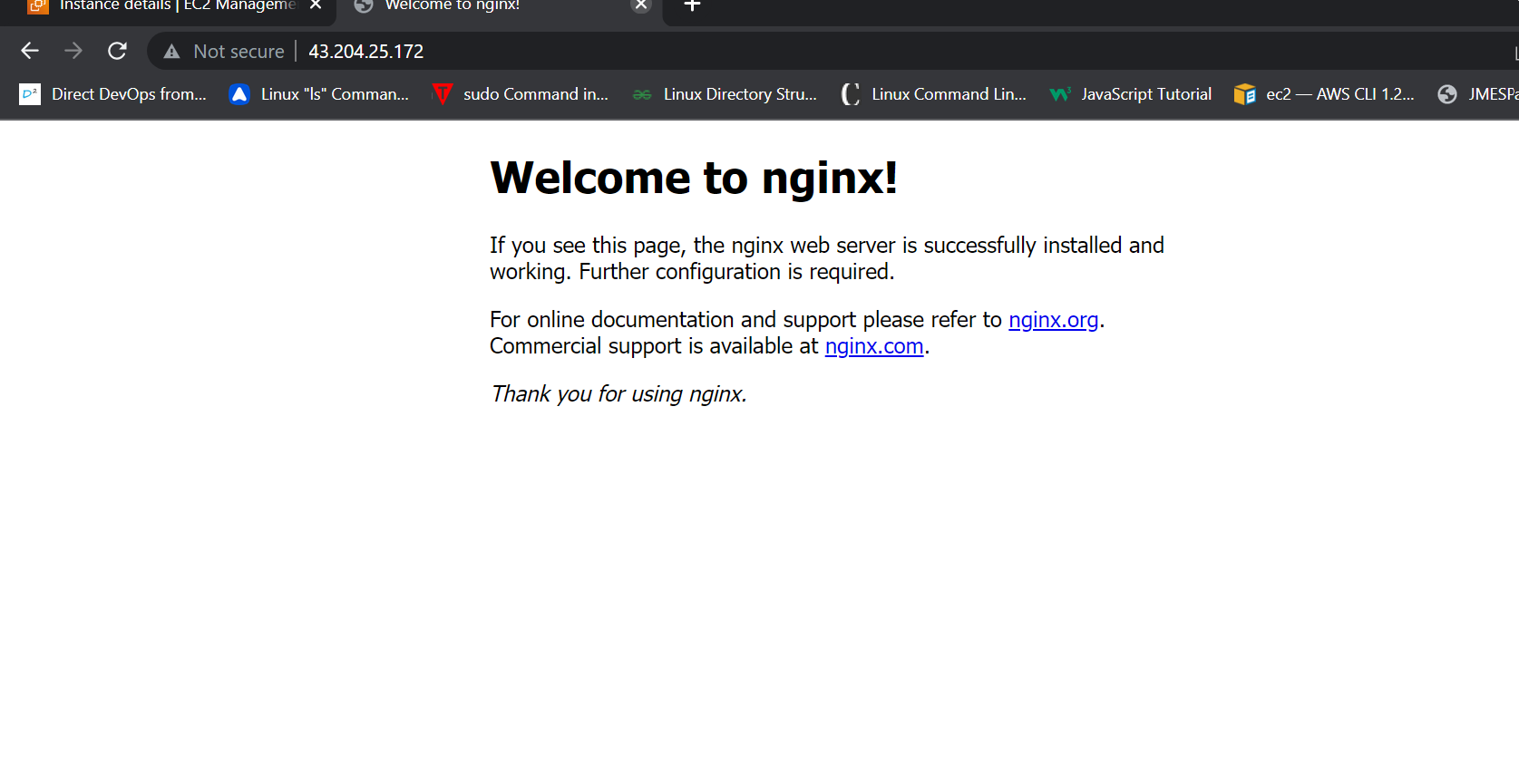
sudo apt-get install nginx

Run the nginx service:

sudo systemctl start nginx

and check its status:

sudo systemctl status nginx



To configure nginx as a reverse proxy to forward requests to your ASP.NET Core app, modify /etc/nginx/sites-available/default. Open it in a text editor and replace the contents with the following: (remove the old content upto default server and replace with below content)

# Default server configuration

#

server {

listen 80 default\_server;

listen [::]:80 default\_server;

server\_name nopCommerce.com;

location / {

proxy\_pass http://localhost:5000;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection keep-alive;

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

# SSL configuration

#

# listen 443 ssl default\_server;

# listen [::]:443 ssl default\_server;

#

# Note: You should disable gzip for SSL traffic.

# See: https://bugs.debian.org/773332

#

# Read up on ssl\_ciphers to ensure a secure configuration.

# See: https://bugs.debian.org/765782

#

# Self signed certs generated by the ssl-cert package

# Don't use them in a production server!

#

# include snippets/snakeoil.conf;

}

Get nopCommerce

Create a directory:

mkdir /var/www/nopCommerce

Download and unpack nopCommerce:

cd /var/www/nopCommerce

sudo wget https://github.com/nopSolutions/nopCommerce/releases/download/release-4.60.2/nopCommerce\_4.60.2\_NoSource\_linux\_x64.zip

sudo apt-get install unzip

sudo unzip nopCommerce\_4.60.2\_NoSource\_linux\_x64.zip

Create couple directories to run nopCommerce:

sudo mkdir bin

sudo mkdir logs

Change the file permissions:

cd ..

sudo chgrp -R www-data nopCommerce/

sudo chown -R www-data nopCommerce/

## Create the nopCommerce service

Create the /etc/systemd/system/nopCommerce.service file with the following contents:

[Unit]

Description=Example nopCommerce app running on Xubuntu

[Service]

WorkingDirectory=/var/www/nopCommerce

ExecStart=/usr/bin/dotnet /var/www/nopCommerce/Nop.Web.dll

Restart=always

# Restart service after 10 seconds if the dotnet service crashes:

RestartSec=10

KillSignal=SIGINT

SyslogIdentifier=nopCommerce-example

User=www-data

Environment=ASPN

ETCORE\_ENVIRONMENT=Production

Environment=DOTNET\_PRINT\_TELEMETRY\_MESSAGE=false

[Install]

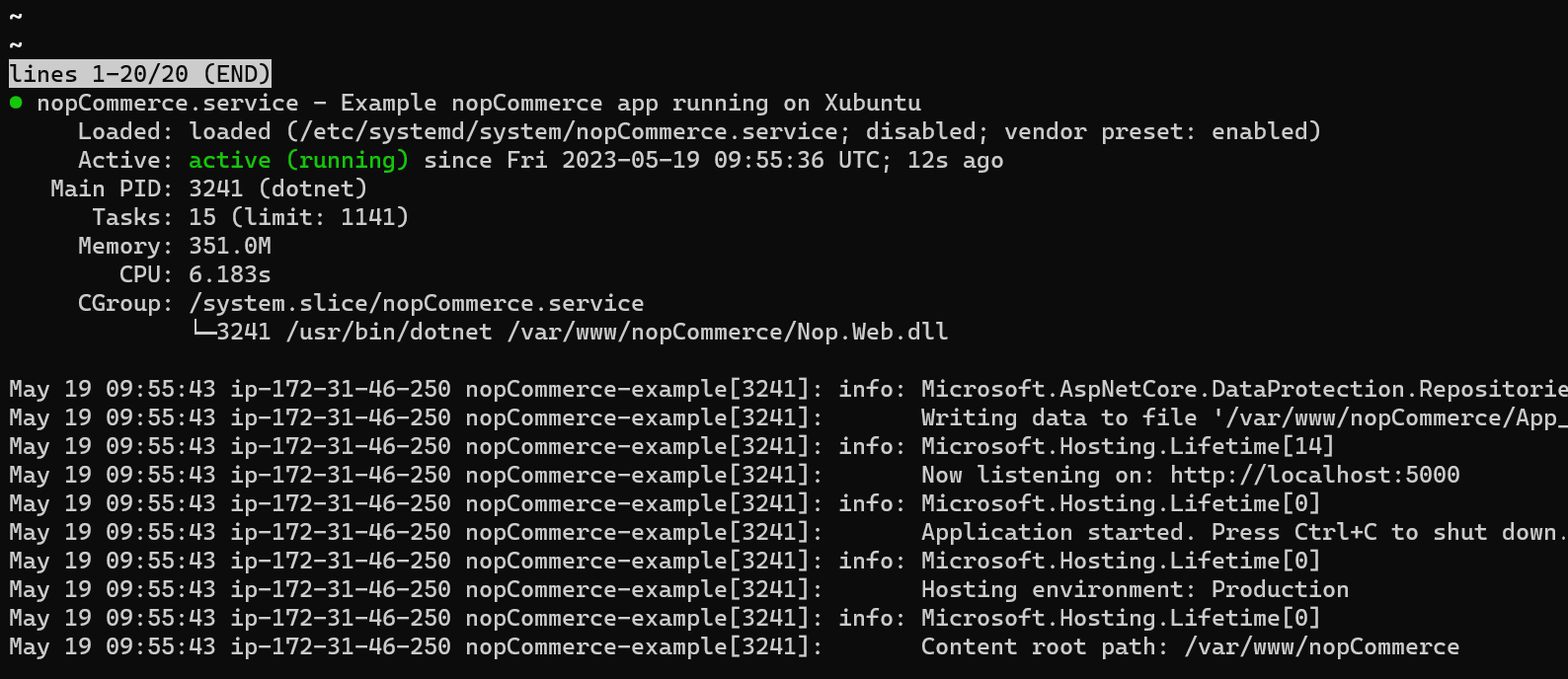
WantedBy=multi-user.target

Start the service:

sudo systemctl start nopCommerce.service

Check the nopCommerce service status:

sudo systemctl status nopCommerce.service



Restart the nginx server:

sudo systemctl restart nginx

now open the browser and check with the ipaddress of the machine.

