TUGAS 1 SISTEM TERDISTRIBUSI

M Raihan Kanjul Adhim

1203210097

1. Buat container microservice1
   # sudo lxc-create --name microservice1 --template download -- --dist "ubuntu" --release "focal" --arch amd64

```
Using image from local cache
Unpacking the rootfs

---
You just created an Ubuntu focal amd64 (20240326_07:42) container.

To enable SSH, run: apt install openssh-server
No default root or user password are set by LXC.
```

2. Buat container microservice2
   # sudo lxc-create --name microservice2 --template download -- --dist "ubuntu" --release "bionic" --arch amd64

```
Using image from local cache
Unpacking the rootfs

---
You just created an Ubuntu bionic amd64 (20240326_07:42) container.

To enable SSH, run: apt install openssh-server
No default root or user password are set by LXC.
```

3. Jalankan container yang sudah di buat tadi
   # sudo lxc-start microservice_1
   # sudo lxc-start microservice_1
   # sudo lxc-ls -f

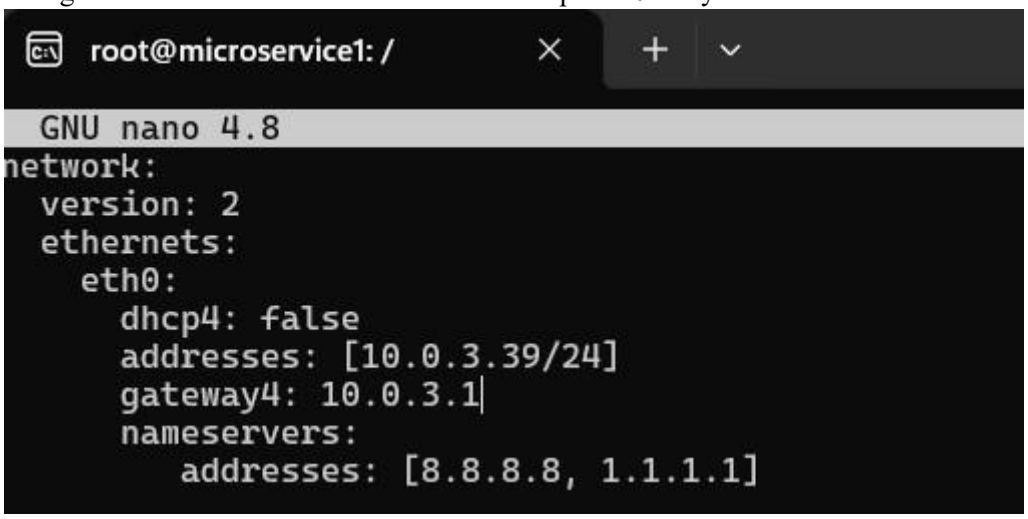
```
NAME           STATE   AUTOSTART GROUPS IPV4       IPV6 UNPRIVILEGED
microservice_1 RUNNING 0         -      10.0.3.39  -    false
microservice_2 RUNNING 0         -      10.0.3.142 -    false
```

4. Setting konfigurasi microservice_1
   # lxc-attach -n microservice1
   # apt update
   # apt install nginx nginx-extras nano net-tools curl

```
root@microservice1:/# apt update
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:2 http://archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2821 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [3200 kB]
Get:6 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [427 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [2699 kB]
Get:8 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [377 kB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [952 kB]
Get:10 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [200 kB]
Get:11 http://archive.ubuntu.com/ubuntu focal-updates/main Translation-en [510 kB]
Get:12 http://archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [2815 kB]
Get:13 http://archive.ubuntu.com/ubuntu focal-updates/restricted Translation-en [392 kB]
Get:14 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [1177 kB]
Get:15 http://archive.ubuntu.com/ubuntu focal-updates/universe Translation-en [282 kB]
Get:16 http://archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 Packages [26.2 kB]
Get:17 http://archive.ubuntu.com/ubuntu focal-updates/multiverse Translation-en [7880 B]
Fetched 16.1 MB in 8s (1890 kB/s)
Reading package lists... Done
```

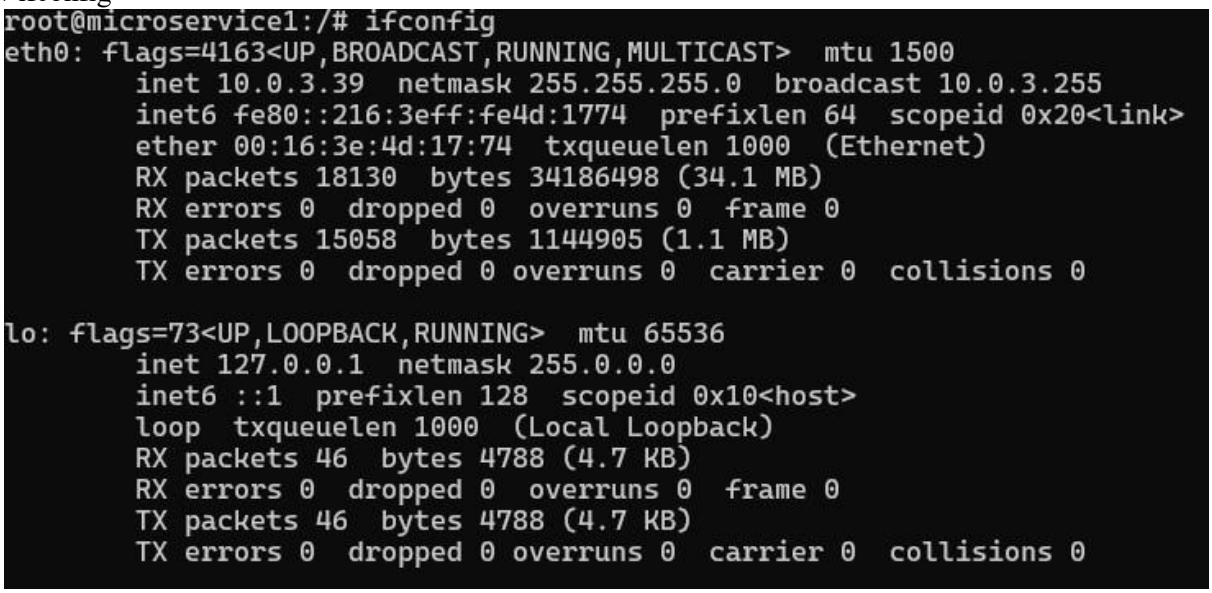
   #exit

5. setting static IP di microservice1 # nano /etc/netplan/10-lxc.yaml



sudo netplan apply

ifconfig



6. setting network interfaces

nano /etc/network/interfaces

interfaces(5) file used by ifup(8) and ifdown(8)

Include files from /etc/network/interfaces.d:

auto lo

iface lo inet loopback

auto eth0 iface eth0 inet

static addresses

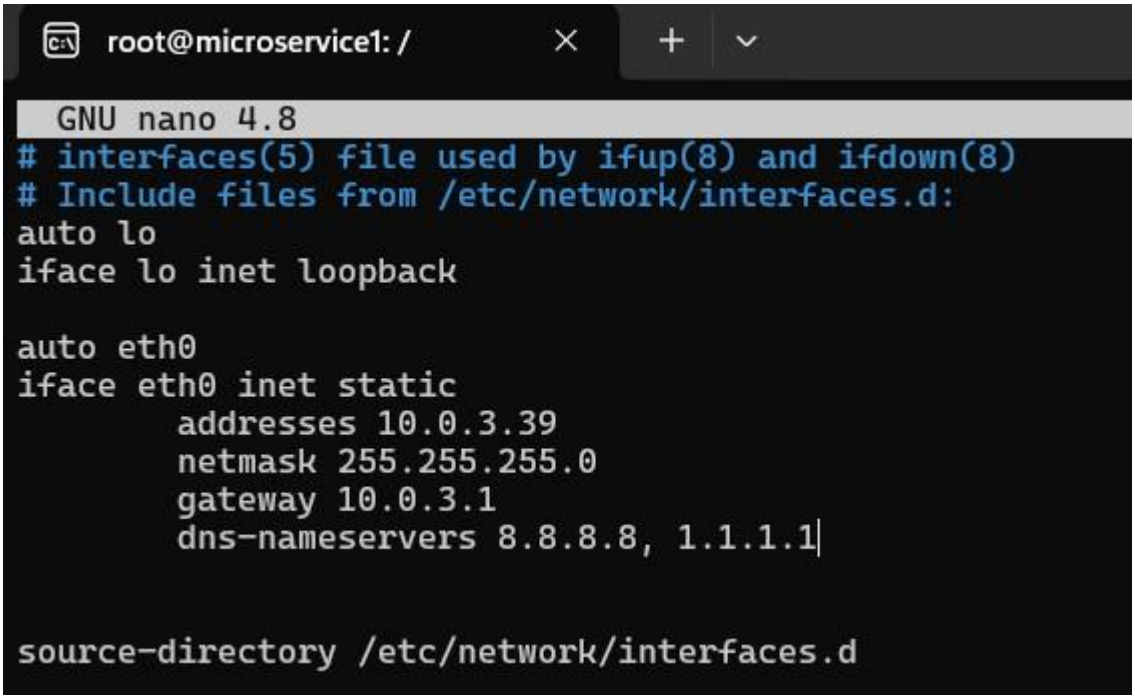
10.0.3.39 netmask

255.255.255.0 gateway

10.0.3.1

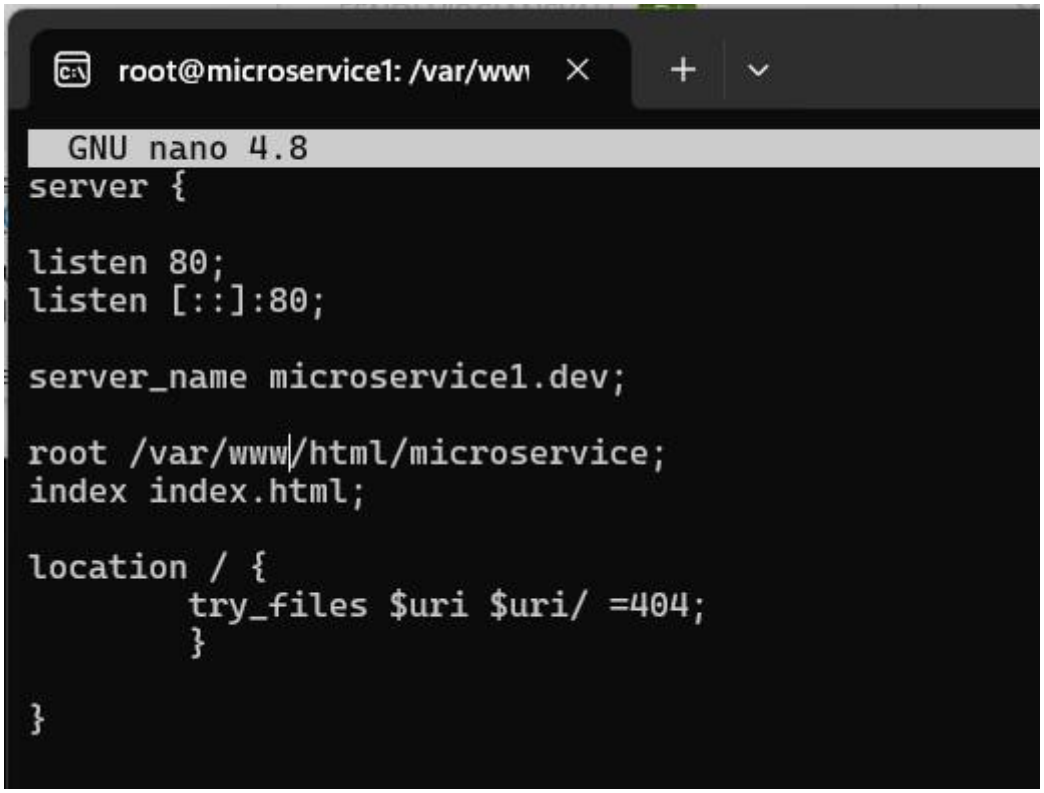
dns-nameservers 8.8.8.8, 1.1.1.1

source-directory /etc/network/interfaces.d

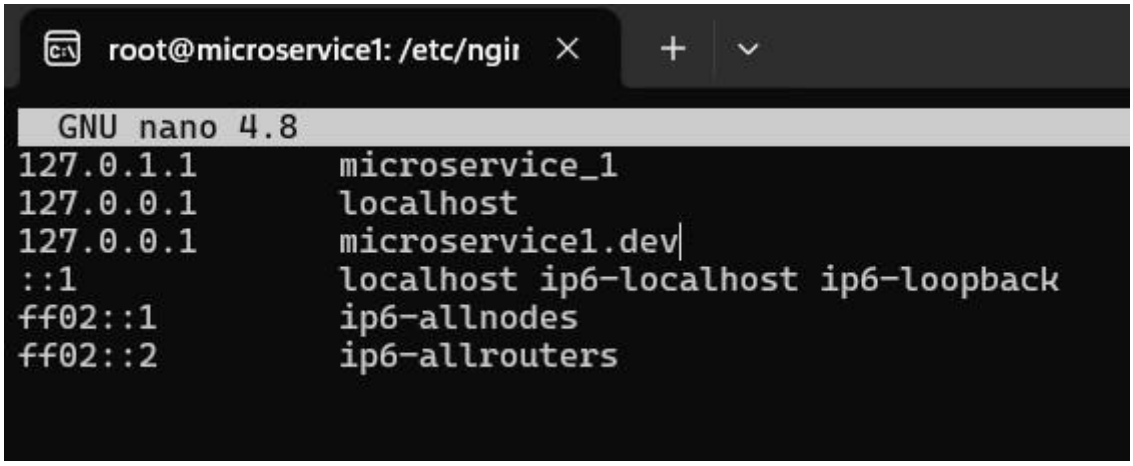


7. Setting ngix

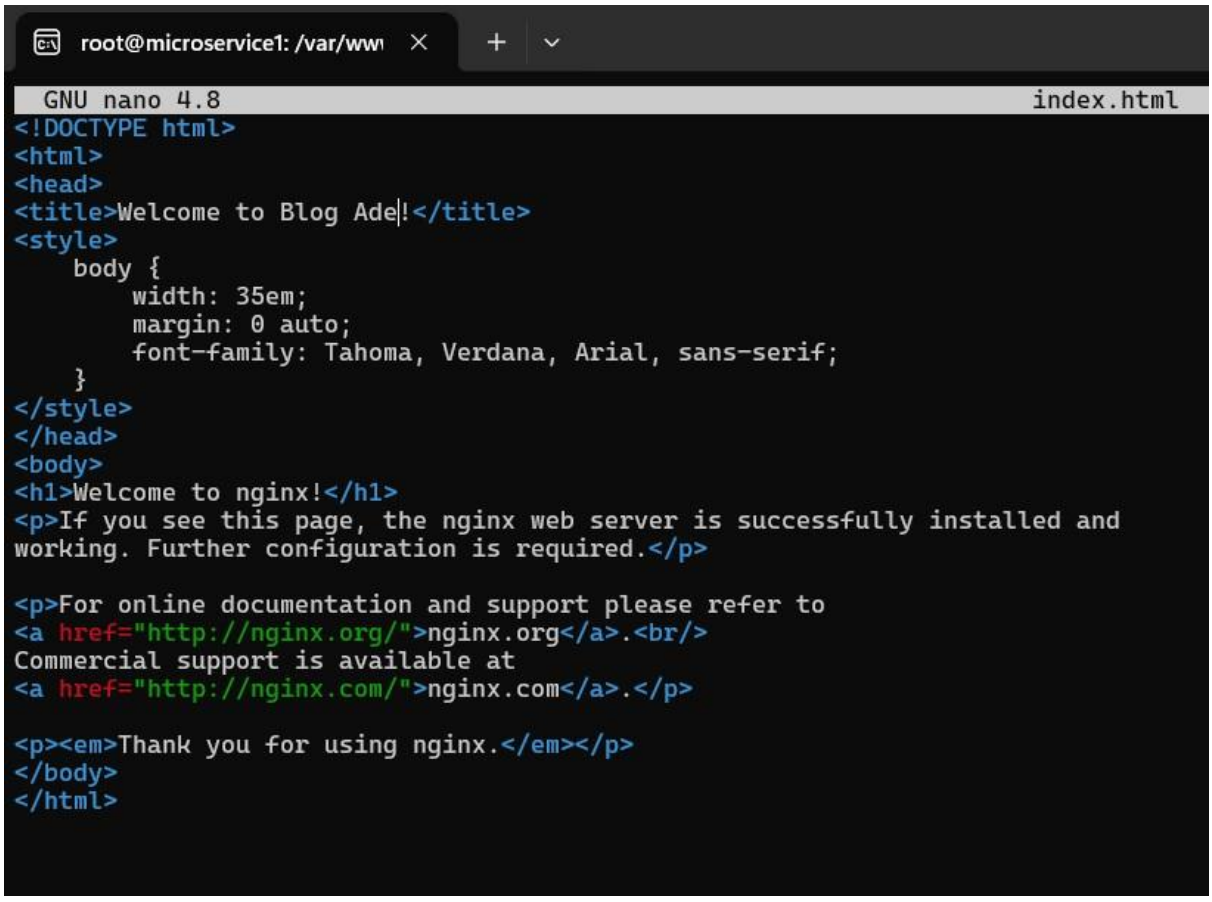
cd /etc/nginx/sites-available
touch microservice1.dev
nano microservice1.dev



cd ../sites-enabled
ln -s /etc/nginx/sites-available/microservice1.dev .
nginx -t
nginx -s reload
nano /etc/hosts



cd /var/www/html && mkdir microservice
cp index.nginx-debian.html microservice/index.html
cd microservice && nano index.html



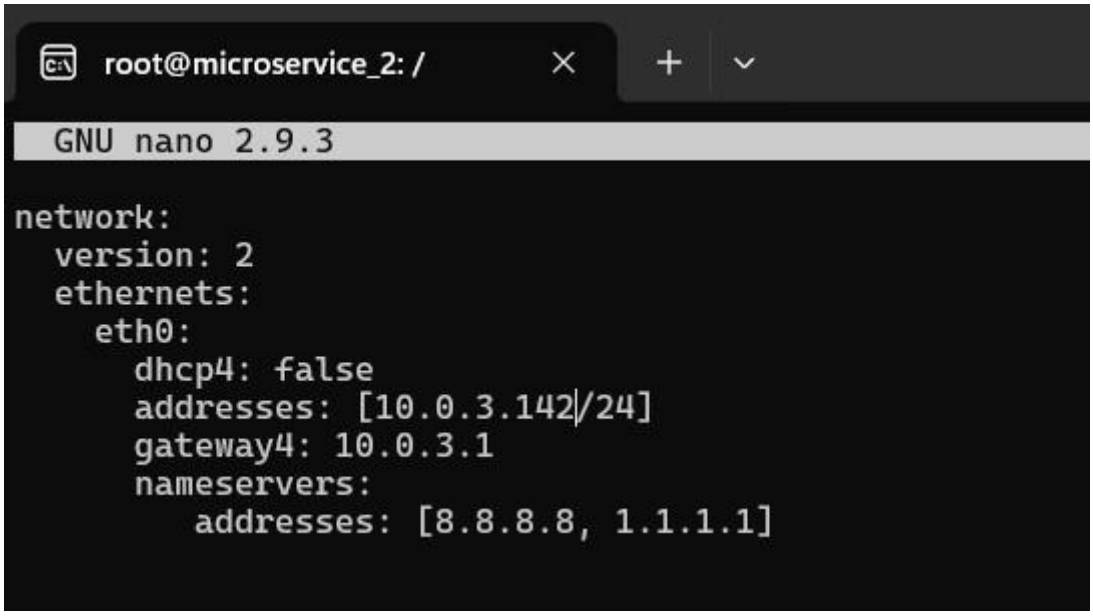
8. Lakukan curl ke microservice1
curl http://microservice1.dev

```
root@microservice1:/var/www/html# curl  http://microservice1.dev
<!DOCTYPE html>
<html>
<head>
<title>Welcome to Blog Ade!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@microservice1:/var/www/html#
```

9. Setting Konfigurasi microservice_2
   # sudo lxc-attach -n microservice_2
   # apt update && apt install nginx ifupdown net-tools nginx-extras nano curl -y

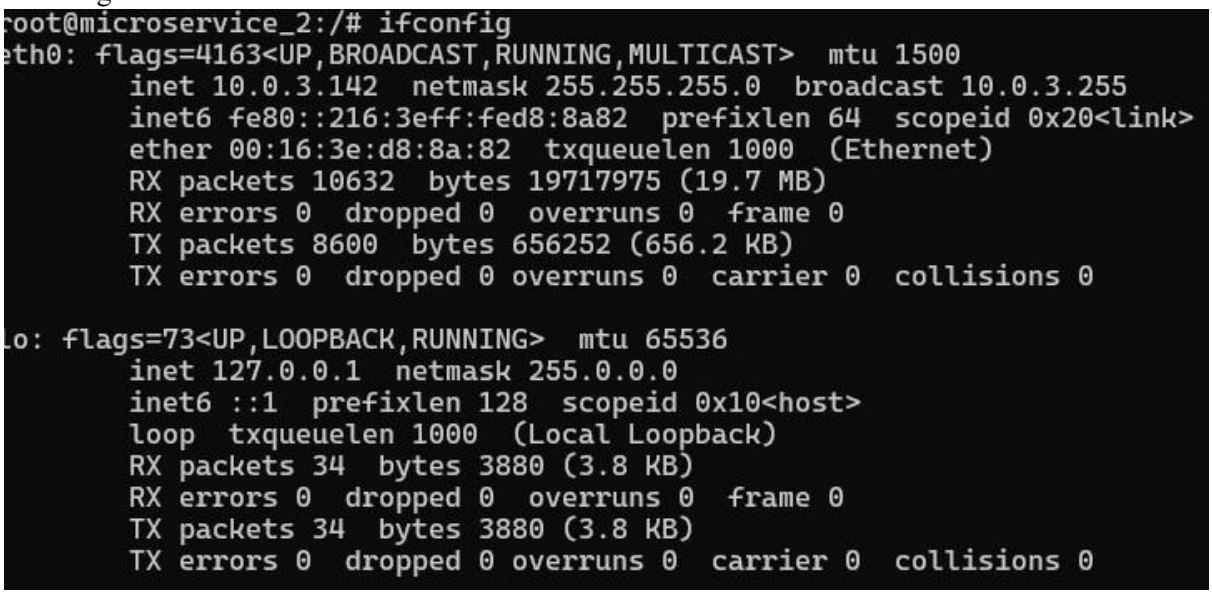
```
[sudo] password for adedian:
root@microservice_2:/# apt update && apt install nginx ifupdown  net-tools nginx-extras nano curl -y
Hit:1 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:2 http://archive.ubuntu.com/ubuntu bionic InRelease
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [3045 kB]
Fetched 3134 kB in 5s (696 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core geoip-database libasn1-8-heimdal libcurl4 libfontconfig1 libfreetype6 libgd3
  libgeoip1 libgssapi3-heimdal libhcrypto4-heimdal libheimbase1-heimdal libheimntlm0-heimdal libhiredis0.13 libhx50
  libjpeg-turbo8 libjpeg8 libkrb5-26-heimdal libldap-2.4-2 libldap-common libluajit-5.1-2 libluajit-5.1-common libn
  libnginx-mod-http-auth-pam libnginx-mod-http-cache-purge libnginx-mod-http-dav-ext libnginx-mod-http-echo libngin
  libnginx-mod-http-geoip libnginx-mod-http-headers-more-filter libnginx-mod-http-image-filter libnginx-mod-http-lu
  libnginx-mod-http-perl libnginx-mod-http-subs-filter libnginx-mod-http-uploadprogress libnginx-mod-http-upstream-
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-nchan libnginx-mod-stream libperl5.26 libpng16-16 li
  librtmp1 libsasl2-2 libsasl2-modules libsasl2-modules-db libtiff5 libwebp6 libwind0-heimdal libxpm4 libxslt1.1 ng
```

   # sudo nano /etc/netplan/10-lxc.yaml



sudo netplan apply

ifconfig



Setting network interfaces

nano /etc/network/interfaces

```
  GNU nano 4.8                    /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:

auto lo
iface lo inet loopback

# primary
auto eth0
iface lo inet static
   address 10.0.3.152
   netmask 255.255.255.0
   gateway 10.0.3.1

source-directory /etc/network/interfaces.d
```

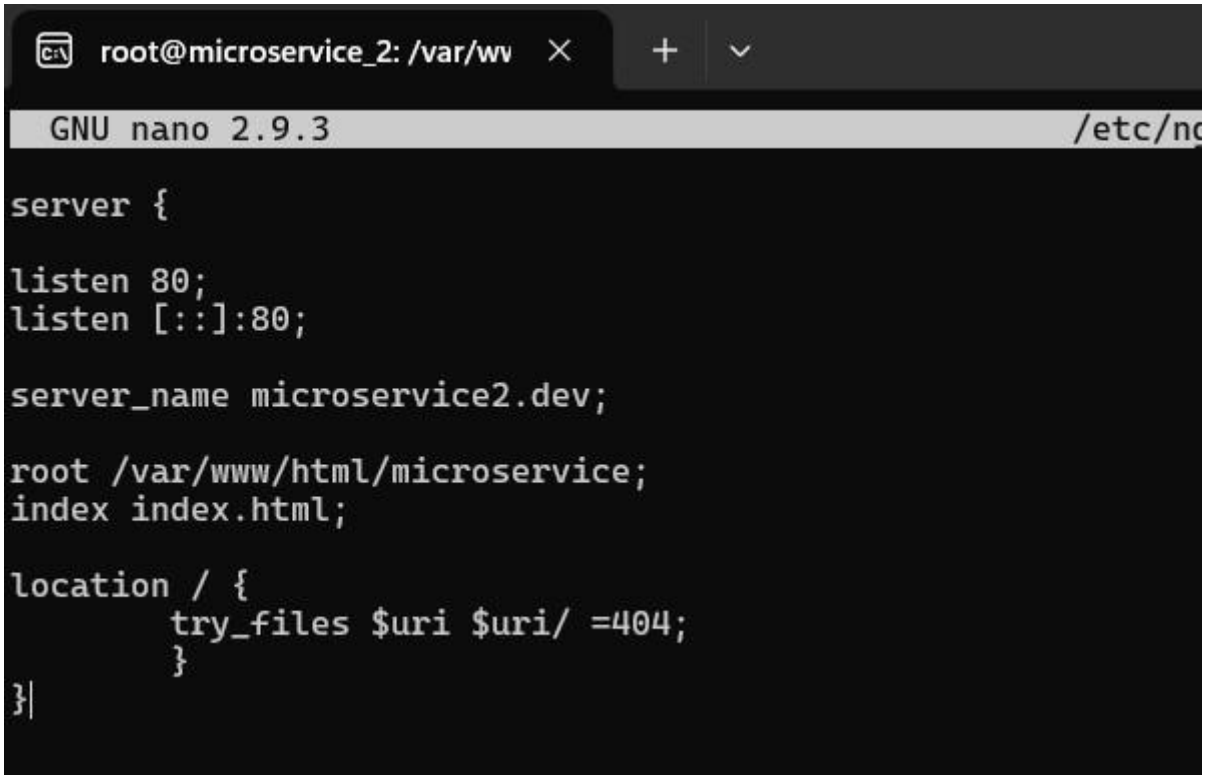
sudo systemctl restart NetworkManager

ifconfig

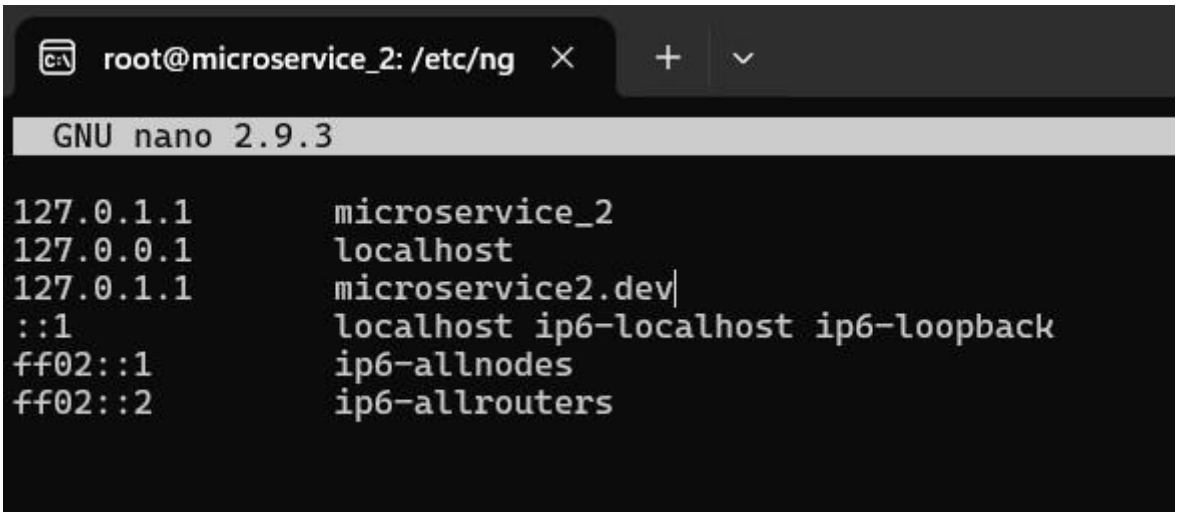
```
root@microservice2:/# sudo systemctl restart NetworkManager
root@microservice2:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.74  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::216:3eff:fef0:9074  prefixlen 64  scopeid 0x20<link>
        ether 00:16:3e:f0:90:74  txqueuelen 1000  (Ethernet)
        RX packets 7040  bytes 26908992 (26.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5723  bytes 441099 (441.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 70  bytes 8044 (8.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 70  bytes 8044 (8.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Setting nginx

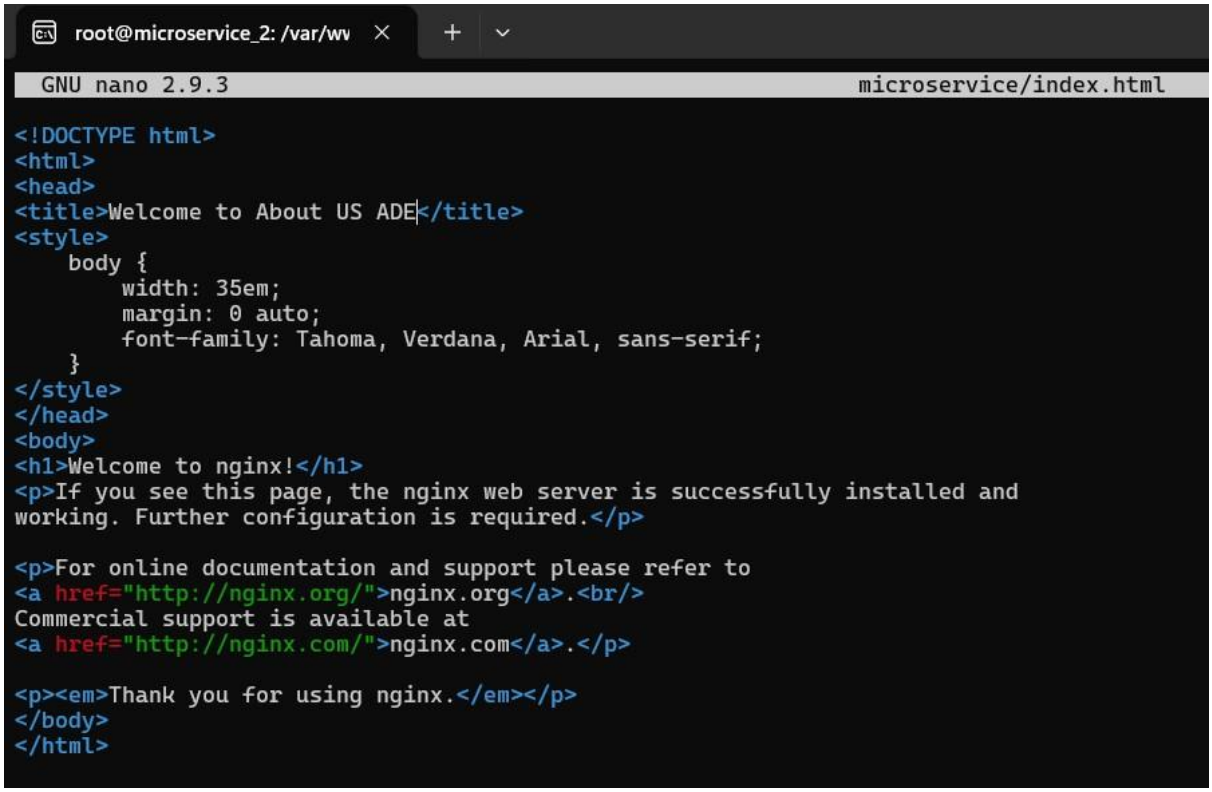
cd /etc/nginx/sites-available && touch microservice2.dev && nano microservice2.dev



cd ../sites-enabled
ln -s /etc/nginx/sites-available/microservice2.dev
nginx -t && nginx -s reload
nano /etc/hosts



cd /var/www/html && mkdir microservice
cp index.nginx-debian.html microservice/index.html && nano microservice2/index.html



curl -i http://microservice2.dev

```
root@microservice_2:/var/www/html# curl -i http://microservice2.dev
HTTP/1.1 200 OK
Server: nginx/1.14.0 (Ubuntu)
Date: Sat, 06 Apr 2024 12:48:20 GMT
Content-Type: text/html
Content-Length: 618
Last-Modified: Sat, 06 Apr 2024 12:46:02 GMT
Connection: keep-alive
ETag: "6611440a-26a"
Accept-Ranges: bytes

<!DOCTYPE html>
<html>
<head>
<title>Welcome to About US ADE</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

10. Setting hosts di WSL Ubuntu 22.04

    # sudo nano /etc/hosts

```
adedian@DESKTOP-3V50F26:

  GNU nano 6.2                                                    /etc/ho
# This file was automatically generated by WSL. To stop automatic generation
# [network]
# generateHosts = false
127.0.0.1       localhost
127.0.1.1       DESKTOP-3V50F26.        DESKTOP-3V50F26
127.0.1.1       sister.local
127.0.1.1       app.sister.local

10.0.3.39       microservice1.dev
10.0.3.142      microservice2.dev


# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
```

    # cd /etc/nginx/sites-available

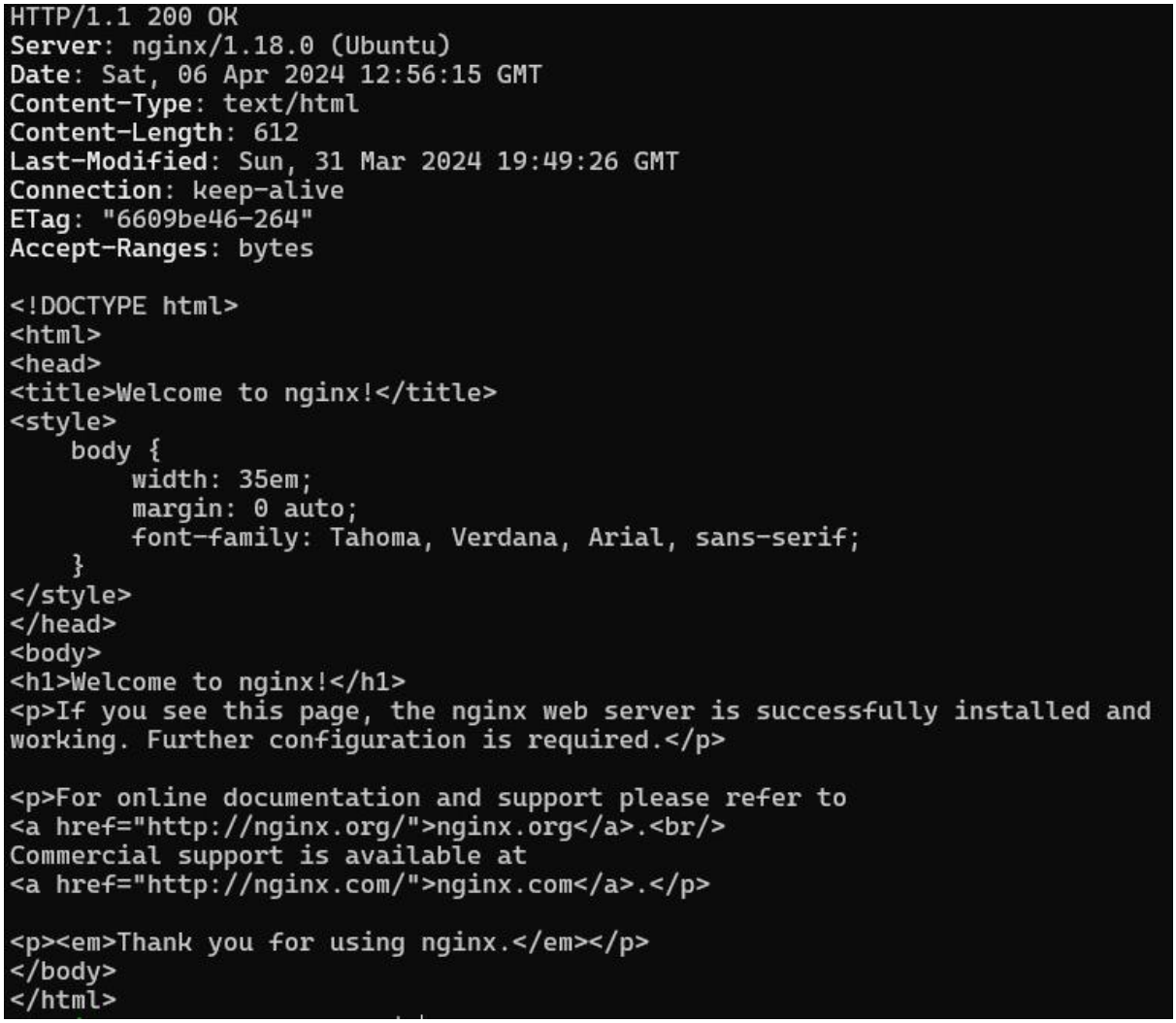
    # touch sister.local

    # nano sister.local

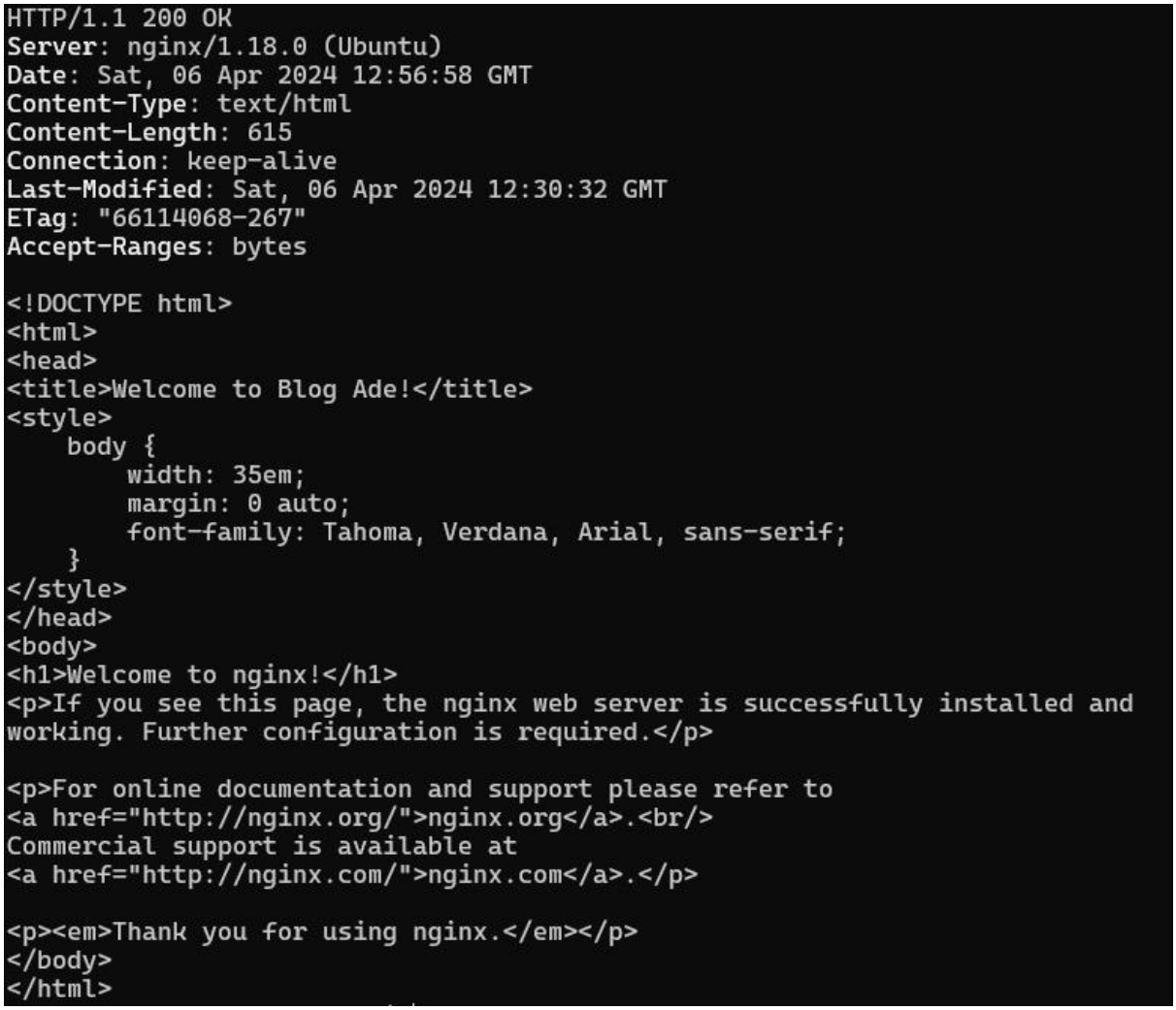
```
server {
        listen 80;
        listen [::]:80;

        server_name sister.local;
        server_name app.sister.local;

        root /var/www/html;
        index index.html;

        location /blog {
                rewrite /blog/?(.*)$ /$1 break;
                proxy_pass http://microservice1.dev;
        }
        location /aboutus {
                rewrite /aboutus/?(.*)$ /$1 break;
                proxy_pass http://microservice2.dev;
        }

        location / {
                try_files $uri $uri/ =404;
}
```

cd ../sites-enabled
sudo ln -s /etc/nginx/sites-available/sister.local .
sudo nginx -t
sudo nginx -s reload

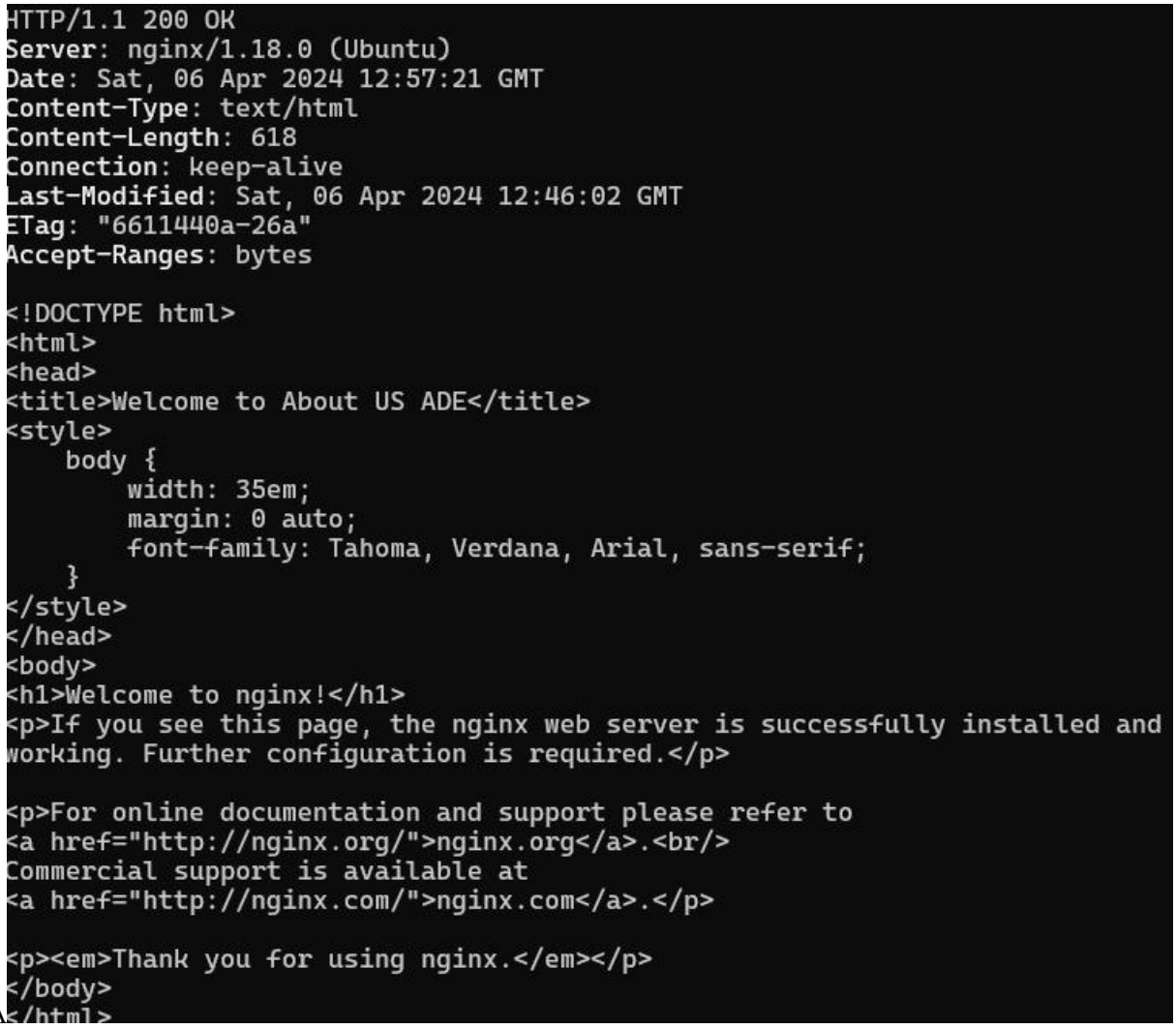
curl -i sister.local



curl -i sister.local/blog



curl -i sister.local/aboutus


HTTP/1.1 200 OK
Server: nginx/1.18.0 (Ubuntu)
Date: Sat, 06 Apr 2024 12:57:21 GMT
Content-Type: text/html
Content-Length: 618
Connection: keep-alive
Last-Modified: Sat, 06 Apr 2024 12:46:02 GMT
ETag: "6611440a-26a"
Accept-Ranges: bytes
<!DOCTYPE html>
<html>
<head>
<title>Welcome to About US ADE</title>
<style>
body {
width: 35em;
margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.
For online documentation and support please refer to
nginx.org.
Commercial support is available at
nginx.com.
Thank you for using nginx.
</body>
</html>
A