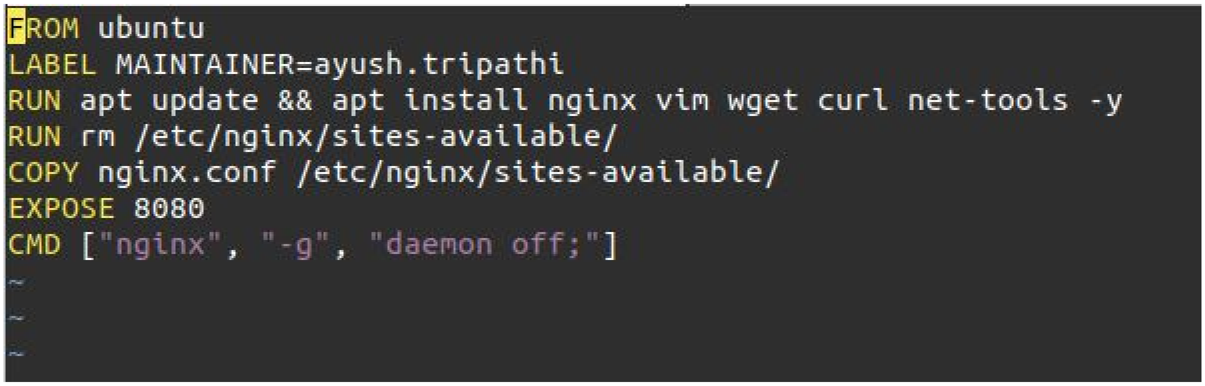
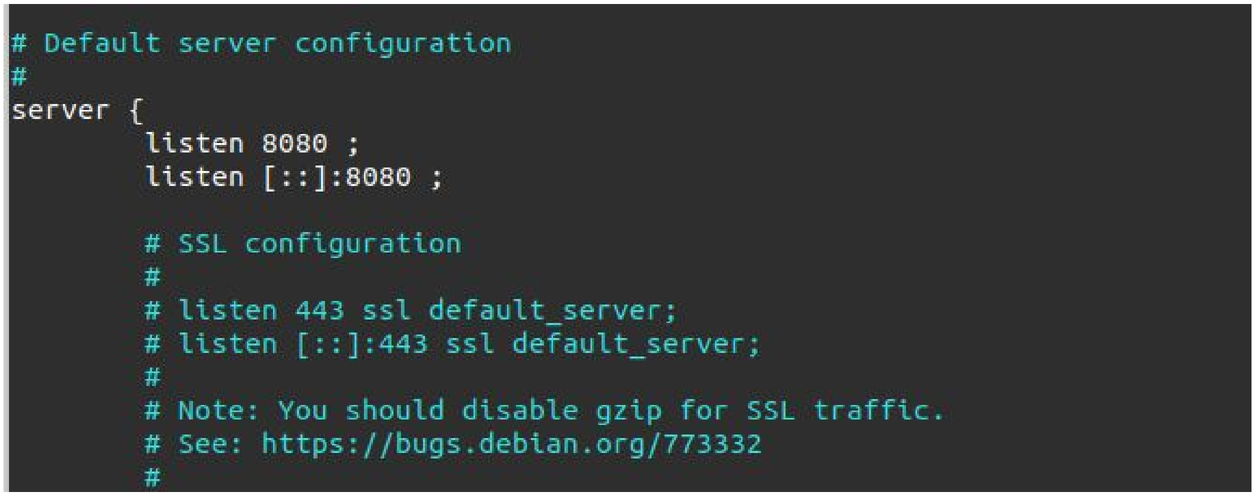
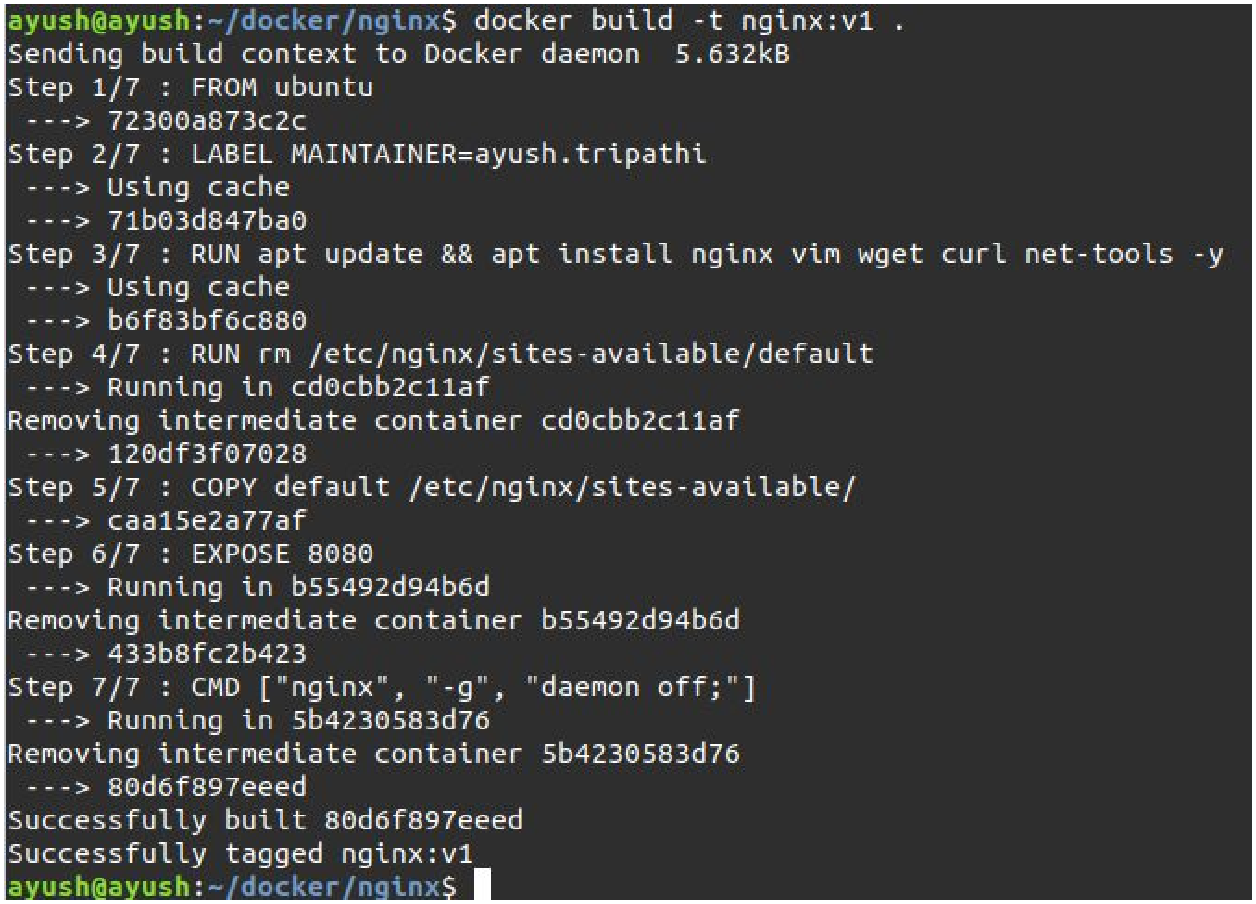
1. **Set the base image to Ubuntu**
    - **Add File Author / Maintainer**
    - **Install Nginx**
    - **Install necessary tools: vim wget curl net-tools**
    - **Remove the default Nginx configuration file**
    - **Copy a configuration file from the current directory**
    - **Expose ports (80)**
    - **Set the default command to execute Nginx when creating a new container**



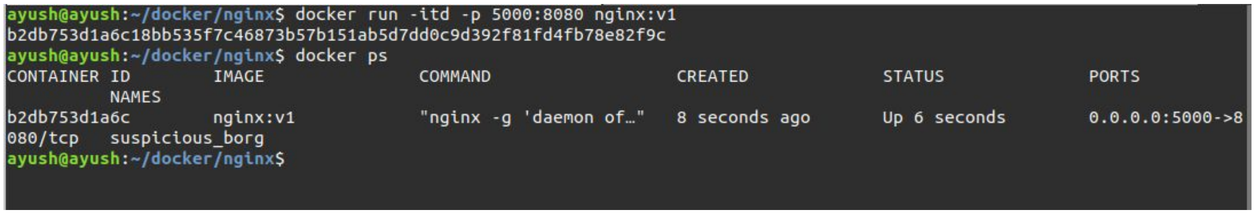
```
FROM ubuntu
LABEL MAINTAINER=ayush.tripathi
RUN apt update && apt install nginx vim wget curl net-tools -y
RUN rm /etc/nginx/sites-available/
COPY nginx.conf /etc/nginx/sites-available/
EXPOSE 8080
CMD ["nginx", "-g", "daemon off;"]
~
~
~
```



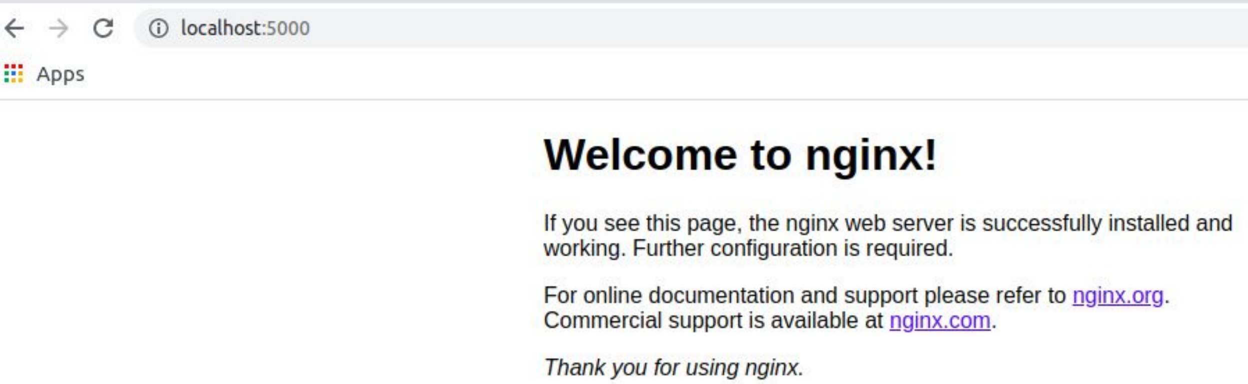
```
# Default server configuration
#
server {
        listen 8080 ;
        listen [::]:8080 ;

        # 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
        #
```



```
ayush@ayush:~/docker/nginx$ docker build -t nginx:v1 .
Sending build context to Docker daemon  5.632kB
Step 1/7 : FROM ubuntu
 ---> 72300a873c2c
Step 2/7 : LABEL MAINTAINER=ayush.tripathi
 ---> Using cache
 ---> 71b03d847ba0
Step 3/7 : RUN apt update && apt install nginx vim wget curl net-tools -y
 ---> Using cache
 ---> b6f83bf6c880
Step 4/7 : RUN rm /etc/nginx/sites-available/default
 ---> Running in cd0cbb2c11af
Removing intermediate container cd0cbb2c11af
 ---> 120df3f07028
Step 5/7 : COPY default /etc/nginx/sites-available/
 ---> caa15e2a77af
Step 6/7 : EXPOSE 8080
 ---> Running in b55492d94b6d
Removing intermediate container b55492d94b6d
 ---> 433b8fc2b423
Step 7/7 : CMD ["nginx", "-g", "daemon off;"]
 ---> Running in 5b4230583d76
Removing intermediate container 5b4230583d76
 ---> 80d6f897eeed
Successfully built 80d6f897eeed
Successfully tagged nginx:v1
ayush@ayush:~/docker/nginx$ 
```



```
ayush@ayush:~/docker/nginx$ docker run -itd -p 5000:8080 nginx:v1
b2db753d1a6c18bb535f7c46873b57b151ab5d7dd0c9d392f81fd4fb78e82f9c
ayush@ayush:~/docker/nginx$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
        NAMES
b2db753d1a6c        nginx:v1            "nginx -g 'daemon of…"   8 seconds ago       Up 6 seconds        0.0.0.0:5000->8
080/tcp   suspicious_borg
ayush@ayush:~/docker/nginx$
```



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

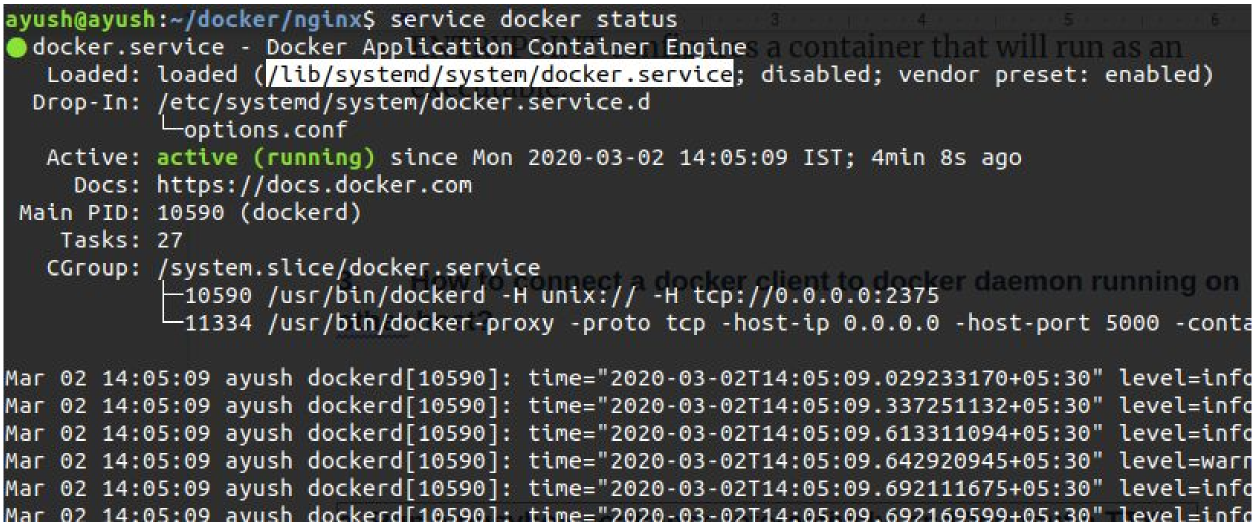
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

2. What is the difference between 'RUN', 'CMD', & 'ENTRYPOINT' in dockerfile?

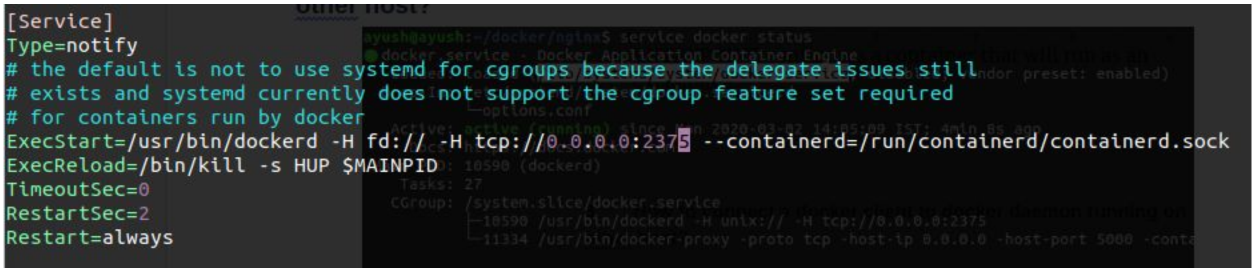
- RUN executes command(s) in a new layer and creates a new image. E.g., it is often used for installing software packages.
- CMD sets default command and/or parameters, which can be overwritten from command line when docker container runs.
- ENTRYPOINT configures a container that will run as an executable.

3. How to connect a docker client to docker daemon running on other host?

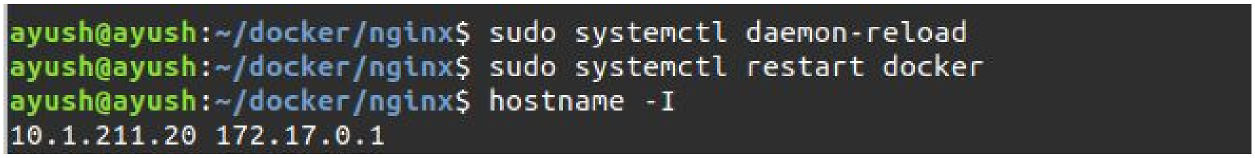


```
ayush@ayush:~/docker/nginx$ service docker status
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; disabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/docker.service.d
           └─options.conf
   Active: active (running) since Mon 2020-03-02 14:05:09 IST; 4min 8s ago
     Docs: https://docs.docker.com
 Main PID: 10590 (dockerd)
    Tasks: 27
   CGroup: /system.slice/docker.service
           ├─10590 /usr/bin/dockerd -H unix:// -H tcp://0.0.0.0:2375
           └─11334 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 5000 -conta

Mar 02 14:05:09 ayush dockerd[10590]: time="2020-03-02T14:05:09.029233170+05:30" level=info
Mar 02 14:05:09 ayush dockerd[10590]: time="2020-03-02T14:05:09.337251132+05:30" level=info
Mar 02 14:05:09 ayush dockerd[10590]: time="2020-03-02T14:05:09.613311094+05:30" level=info
Mar 02 14:05:09 ayush dockerd[10590]: time="2020-03-02T14:05:09.642920945+05:30" level=warn
Mar 02 14:05:09 ayush dockerd[10590]: time="2020-03-02T14:05:09.692111675+05:30" level=info
Mar 02 14:05:09 ayush dockerd[10590]: time="2020-03-02T14:05:09.692169599+05:30" level=info
```

Open that file

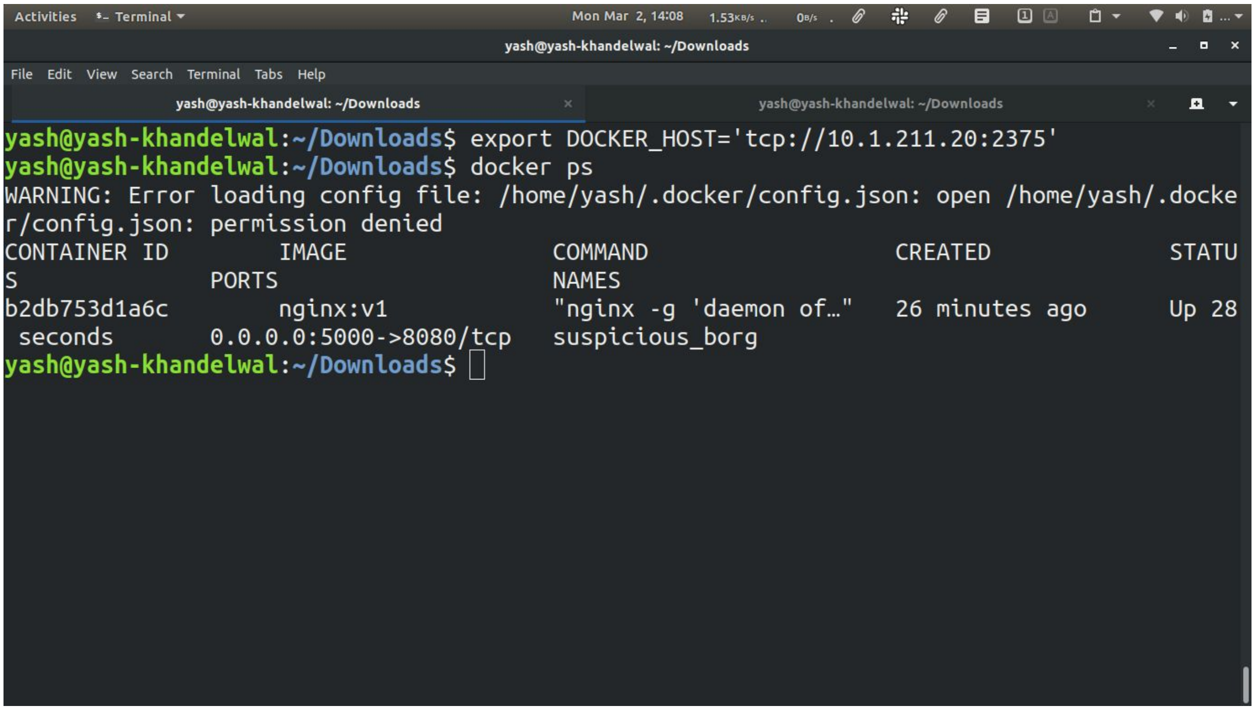


```
[Service]
Type=notify
# the default is not to use systemd for cgroups because the delegate issues still
# exists and systemd currently does not support the cgroup feature set required
# for containers run by docker
ExecStart=/usr/bin/dockerd -H fd:// -H tcp://0.0.0.0:2375 --containerd=/run/containerd/containerd.sock
ExecReload=/bin/kill -s HUP $MAINPID
TimeoutSec=0
RestartSec=2
Restart=always
```



```
ayush@ayush:~/docker/nginx$ sudo systemctl daemon-reload
ayush@ayush:~/docker/nginx$ sudo systemctl restart docker
ayush@ayush:~/docker/nginx$ hostname -I
10.1.211.20 172.17.0.1
```

From another host



Activities
Terminal
Mon Mar 2, 14:08
1.53KB/s
0B/s
yash@yash-khandelwal: ~/Downloads
File
Edit
View
Search
Terminal
Tabs
Help
yash@yash-khandelwal: ~/Downloads
yash@yash-khandelwal: ~/Downloads

```
yash@yash-khandelwal:~/Downloads$ export DOCKER_HOST='tcp://10.1.211.20:2375'
yash@yash-khandelwal:~/Downloads$ docker ps
WARNING: Error loading config file: /home/yash/.docker/config.json: open /home/yash/.docke
r/config.json: permission denied
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATU
S                PORTS                    NAMES
b2db753d1a6c        nginx:v1            "nginx -g 'daemon of…"   26 minutes ago      Up 28
 seconds         0.0.0.0:5000->8080/tcp   suspicious_borg
yash@yash-khandelwal:~/Downloads$ 
```