1. Problem 1:

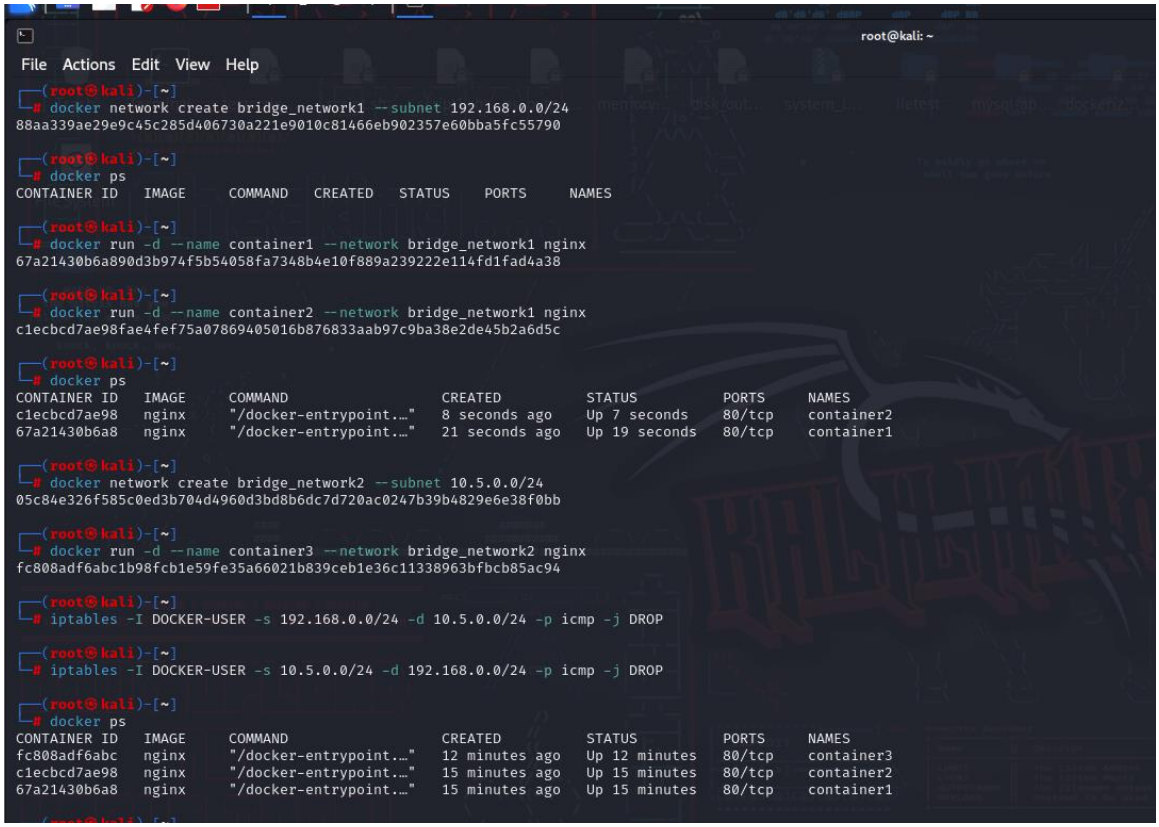
Create bridge network with subnet 192.168.0.0/24.

Run 2 containers and attach containers to this network.

Create another bridge network with subnet 10.5.0.0/24.

Run any container and attach it to the new network.

Make sure that the containers at different network can’t ping each other



2. Problem 2:

Create static html file

Write Dockerfile to build image based on httpd to host the html file and

specify the following

Copy the html file.

Copy a new configuration file to listen on port 9999 instead of 80

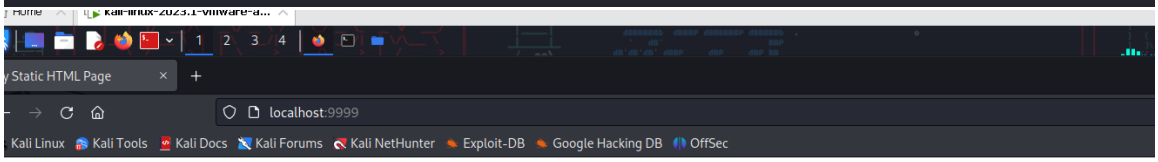
Open the port 9999 in the container

Add environment variable CONTAINER with value docker .

Add startup command to echo the variable

```
┌──(root㉿kali)-[/home/kali/Desktop/Docker3]
└─# vim Dockerfile

┌──(root㉿kali)-[/home/kali/Desktop/Docker3]
└─# cat Dockerfile
FROM httpd:latest

COPY index.html /usr/local/apache2/htdocs/

COPY httpd.conf /usr/local/apache2/conf/httpd.conf

EXPOSE 9999

ENV CONTAINER=docker

CMD echo "The value of CONTAINER is: $CONTAINER"


┌──(root㉿kali)-[/home/kali/Desktop/Docker3]
└─# vim httpd.conf

┌──(root㉿kali)-[/home/kali/Desktop/Docker3]
└─# cat httpd.conf
Listen 9999
```

```
┌──(root㉿kali)-[/home/kali/Desktop/Docker3]
└─# docker build -t my-httpd-image .
Sending build context to Docker daemon  4.096kB
Step 1/6 : FROM httpd:latest
latest: Pulling from library/httpd
f03b40093957: Already exists
abaf8619eb1c: Pull complete
e3fe37d0c2ad: Pull complete
52a1e37affe5: Pull complete
49d8a68fd903: Pull complete
Digest: sha256:1bb3f7669a85713906e695599d29c58ab40d4e6409907946609d92a428e95b49
Status: Downloaded newer image for httpd:latest
 ---> d1676199e605
Step 2/6 : COPY index.html /usr/local/apache2/htdocs/
 ---> 22a83737eed7
Step 3/6 : COPY httpd.conf /usr/local/apache2/conf/httpd.conf
 ---> a180eb07ffe7
Step 4/6 : EXPOSE 9999
 ---> Running in 8eda0f11196d
Removing intermediate container 8eda0f11196d
 ---> 8330d114a429
Step 5/6 : ENV CONTAINER=docker
 ---> Running in 66da1fe63c70
Removing intermediate container 66da1fe63c70
 ---> 22a9e0df183d
Step 6/6 : CMD echo "The value of CONTAINER is: $CONTAINER"
 ---> Running in c9b1cddb1d8b
Removing intermediate container c9b1cddb1d8b
 ---> 514b7d0d7b43
Successfully built 514b7d0d7b43
Successfully tagged my-httpd-image:latest

┌──(root㉿kali)-[/home/kali/Desktop/Docker3]
└─# docker run -d -p 9999:9999 my-httpd-image

9df5e3ef7525c6e693ec07555698680449dfd864088d180bc2fe5bbe9e5cba0d
```



Welcome to my static HTML page!

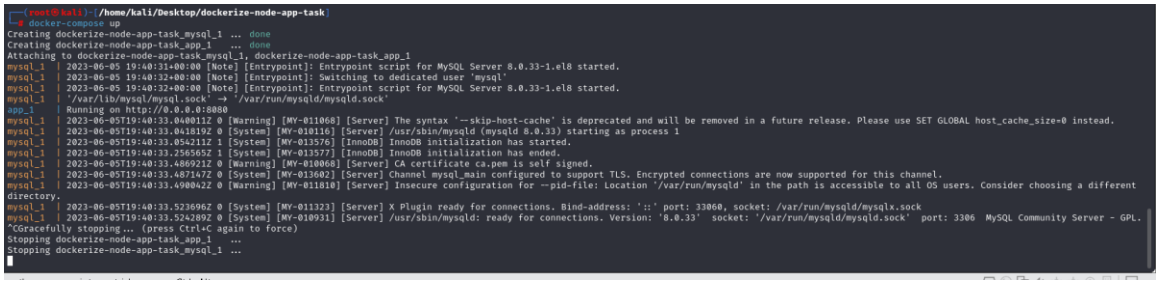
3. Problem 3:

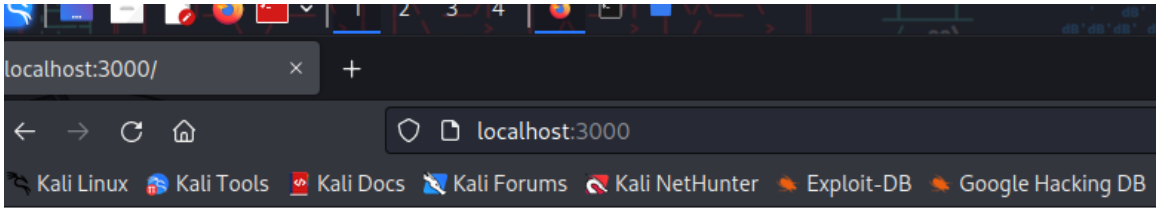
create a docker compose to up mysql container, and

https://github.com/sabreensalama/dockerize-node-app-task which depend on mysqldb.

Add volume for mysqldb

```
┌──(root㉿kali)-[/home/kali/Desktop/dockerize-node-app-task]
└─# cat docker-compose.yml
version: '3'
services:
  mysql:
    image: mysql:latest
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: passwd
      MYSQL_DATABASE: mydbpw
      MYSQL_USER: mohamed
      MYSQL_PASSWORD: mohamed
    volumes:
      - mysql_data:/var/lib/mysql
  app:
    build: .
    restart: always
    depends_on:
      - mysql
    ports:
      - "3000:8080"
volumes:
  mysql_data:
```

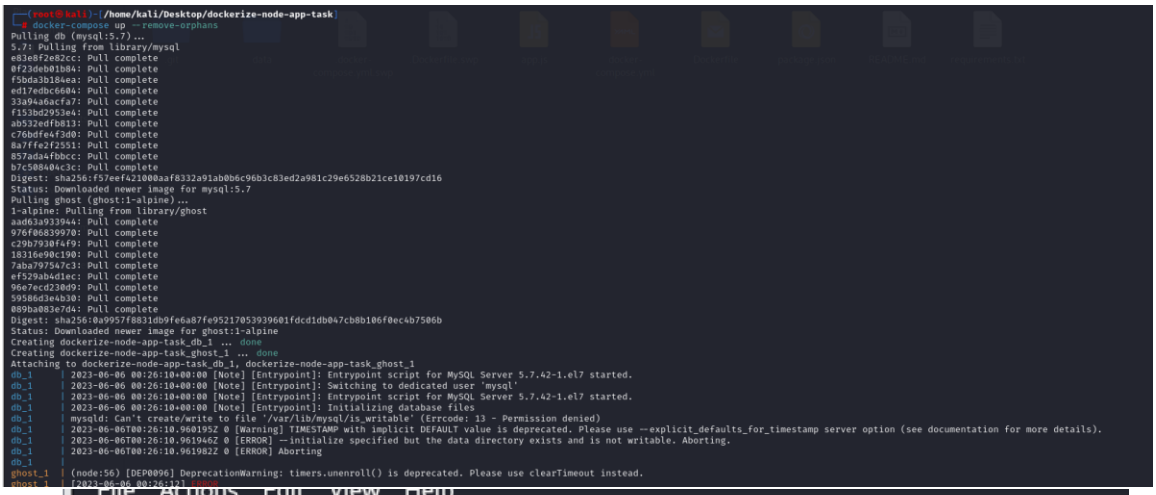


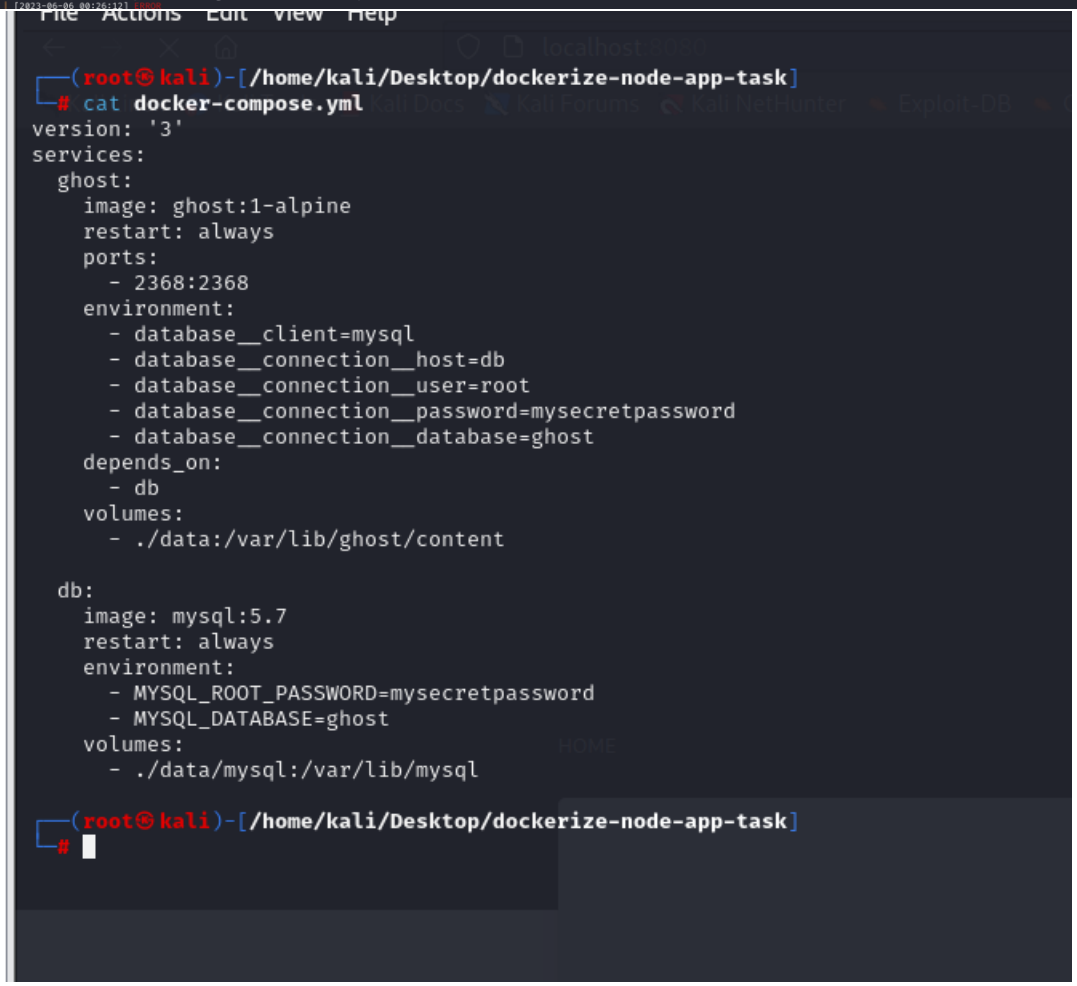


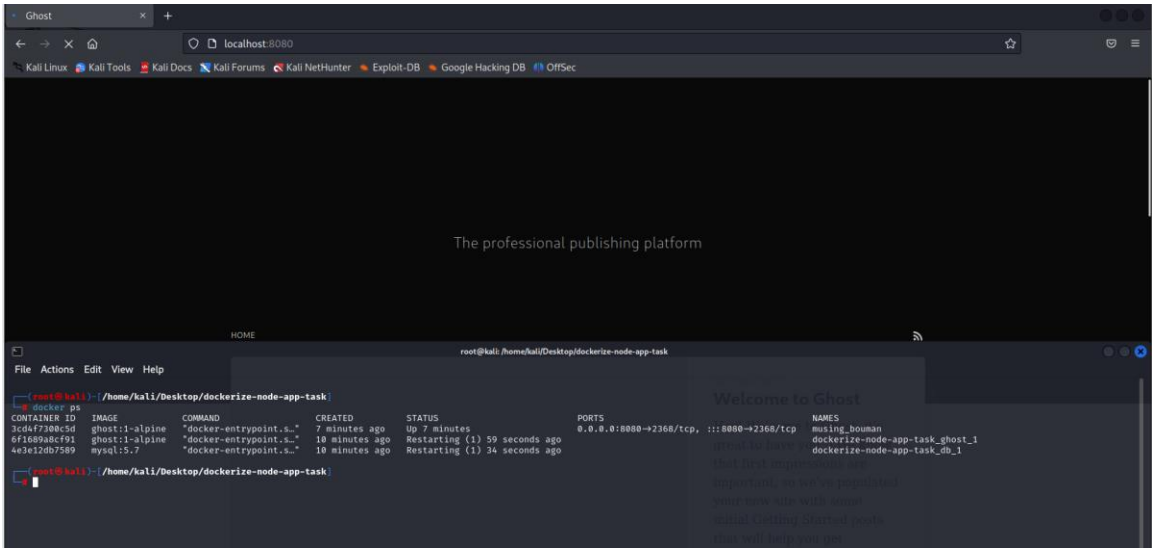
Hello World

4. Problem 4:

Use docker compose to deploy ghost platform (image: ghost:1-alpine)(Ghost is a free and open source blogging platform written in JavaScript) Use mysql database instead of sqlite






Ghost
localhost:8080
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
The professional publishing platform
HOME
root@kali: /home/kali/Desktop/dockerize-node-app-task
File Actions Edit View Help
(root@kali)-[/home/kali/Desktop/dockerize-node-app-task]
docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
3cd4f7308c5d ghost:1-alpine "docker-entrypoint.s…" 7 minutes ago Up 7 minutes 0.0.0.0:8080->2368/tcp, :::8080->2368/tcp muling_bouman
6f1689a8cf91 ghost:1-alpine "docker-entrypoint.s…" 10 minutes ago Restarting (1) 59 seconds ago dockerize-node-app-task_ghost_1
4e3e12db7589 mysql:5.7 "docker-entrypoint.s…" 10 minutes ago Restarting (1) 34 seconds ago dockerize-node-app-task_db_1
(root@kali)-[/home/kali/Desktop/dockerize-node-app-task]
#
Welcome to Ghost