Reverse Proxy using Traefik on Docker compose

The main approach is to run everything using docker compose so there is no need to install any additional stuff apart from docker and docker compose.

- Create a basic reverse proxy set-up using docker based on Traefik official documentation for docker-compose and add the traefik basic service

docker-compose.yml

```yaml
version: "3.3"

services:
  traefik:
    image: "traefik:v2.6"
    command:
      #- "--log.level=DEBUG"
      - "--api.insecure=true"
      - "--providers.docker=true"
      - "--providers.docker.exposedbydefault=false"
      - "--entrypoints.web.address=:80"
    ports:
      - "80:80"
      - "8080:8080"
    volumes:
      - "/var/run/docker.sock:/var/run/docker.sock:ro"

  whoami:
    image: "traefik/whoami"
    labels:
      - "traefik.enable=true"
      - "traefik.http.routers.whoami.rule=Host(`localhost`)"
      - "traefik.http.routers.whoami.entrypoints=web"
```

- Run the docker-compose file

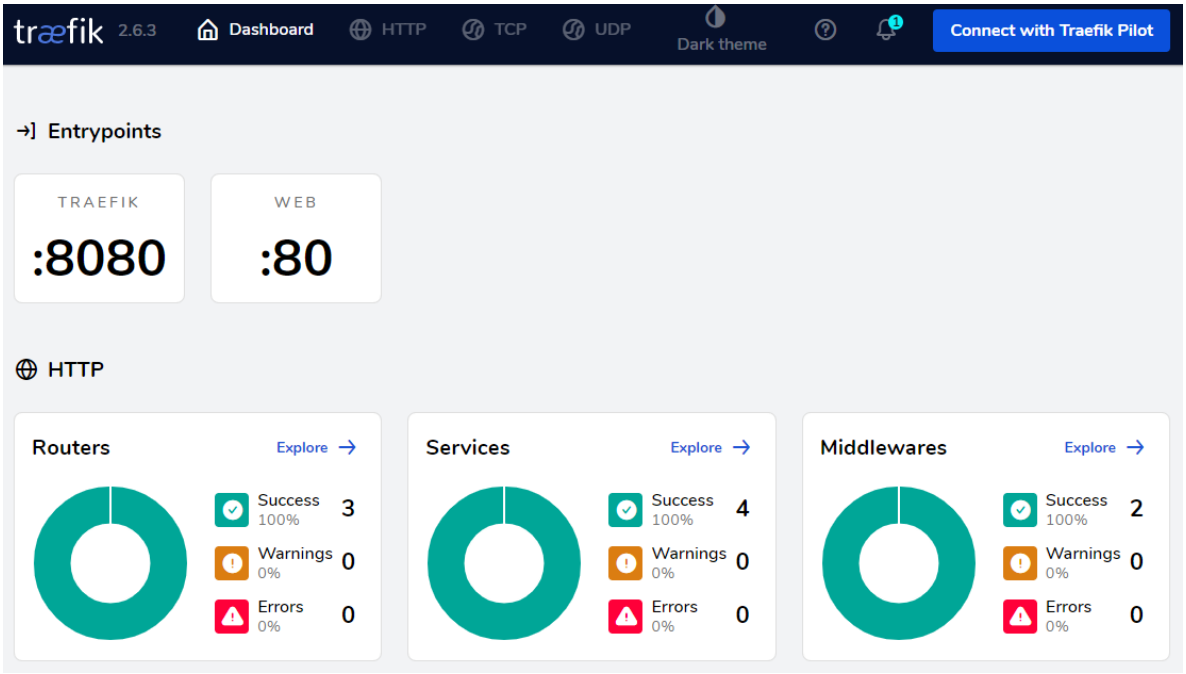
✓ docker-compose up -d

Ubuntu-18.04

```
cris@DESKTOP-PM304DL:~/reverseproxy_traefik$ docker-compose up -d
Pulling traefik (traefik:v2.6)...
v2.6: Pulling from library/traefik
8663204ce13b: Pull complete
1a6b5dadc224: Pull complete
b36d84ad4ebf: Pull complete
3d6b8db5964f: Pull complete
Digest: sha256:126443503c12ced877f806cad0c7bd82ea1fce5d5ff7ac8663c99cede85e961f
Status: Downloaded newer image for traefik:v2.6
Pulling whoami (traefik/whoami:)...
latest: Pulling from traefik/whoami
0fd410c59e5b: Pull complete
911aea188600: Pull complete
417361c31757: Pull complete
Digest: sha256:8d0f943abdbfaa16bad1f858d5bac621ba5ca678c496f278e7bebb8457b82e4e
Status: Downloaded newer image for traefik/whoami:latest
Creating reverseproxy_traefik_traefik_1 ... done
Creating reverseproxy_traefik_whoami_1  ... done
```

- Verify service is up and running on localhost default port (80) as assigned in traefik ports configuration

- Verify traefik dashboard on localhost assigned port (8080)



- Add a second service to the docker-compose file and attach it to traefik

docker-compose.yml

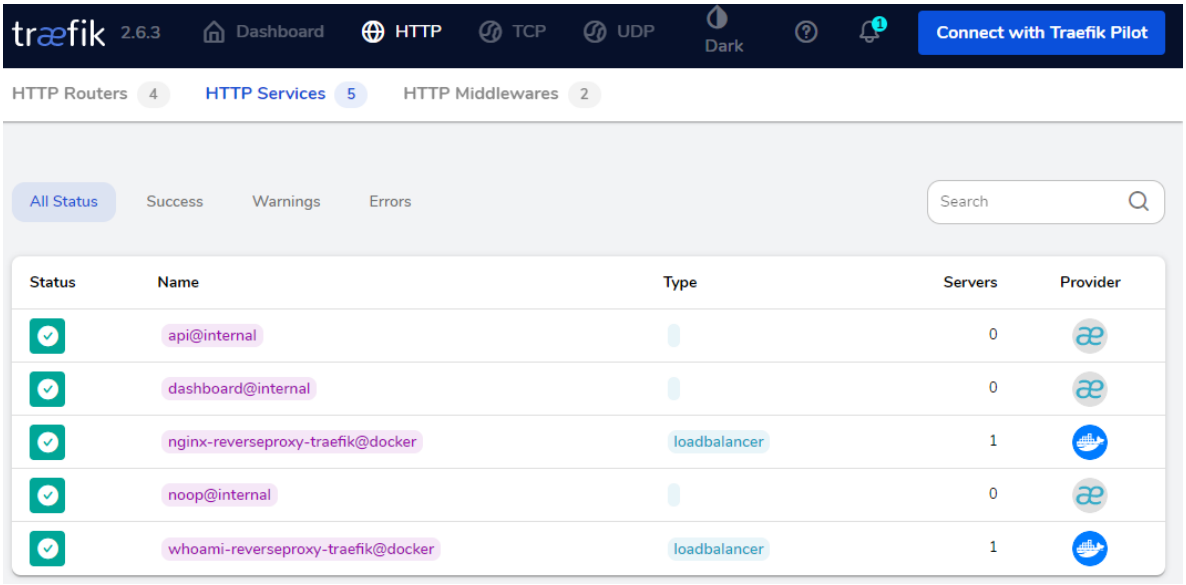
```yaml
version: "3.3"

services:
  traefik:
    image: "traefik:v2.6"
    command:
      #- "--log.level=DEBUG"
      - "--api.insecure=true"
      - "--providers.docker=true"
      - "--providers.docker.exposedbydefault=false"
      - "--entrypoints.web.address=:80"
    ports:
      - "80:80"
      - "8080:8080"
    volumes:
      - "/var/run/docker.sock:/var/run/docker.sock:ro"

  whoami:
    image: "traefik/whoami"
    labels:
      - "traefik.enable=true"
      - "traefik.http.routers.whoami.rule=Host(`localhost`)"
      - "traefik.http.routers.whoami.entrypoints=web"
  nginx:
    image: "nginxdemos/hello"
    labels:
      - "traefik.enable=true"
      - "traefik.http.routers.nginx.rule=Host(`localhost`) && Path(`/service2`)"
      - "traefik.http.routers.nginx.entrypoints=web"
```

- Run the docker-compose file so the container gets updated

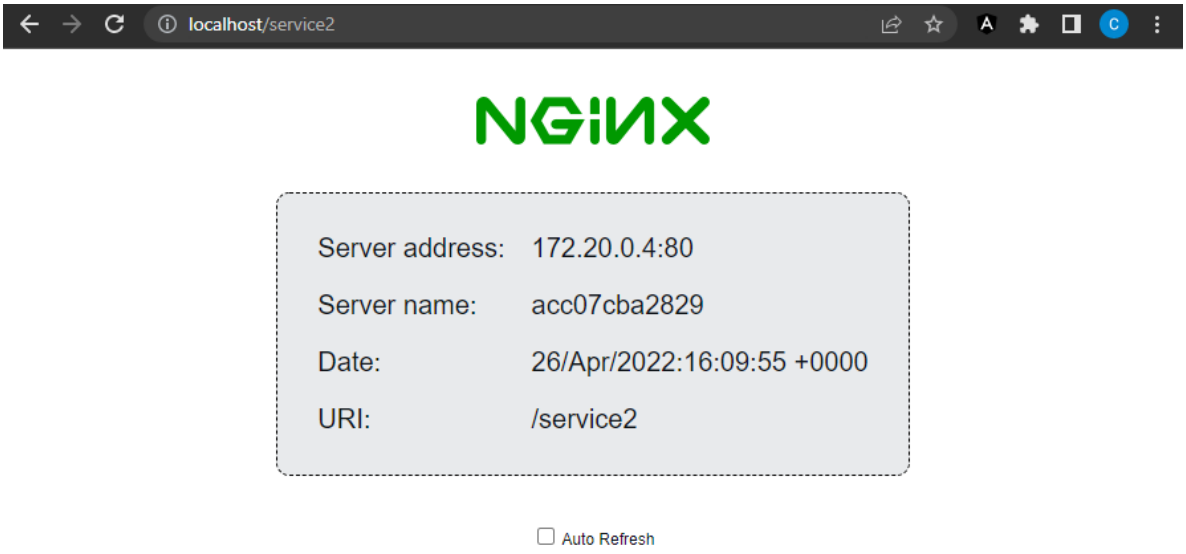
✓ docker-compose up -d

```
cris@DESKTOP-PM304DL:~/reverseproxy_traefik$ docker-compose up -d
Pulling nginx (nginxdemos/hello:)...
latest: Pulling from nginxdemos/hello
df9b9388f04a: Already exists
5867cba5fcbd: Pull complete
4b639e65cb3b: Pull complete
061ed9e2b976: Pull complete
bc19f3e8eeb1: Pull complete
4071be97c256: Pull complete
25e4add94632: Pull complete
b4fee94e4ffe: Pull complete
7ee50b562996: Pull complete
Digest: sha256:e6cac6664be21601899c880addb538c62e7f11ce396c3030254363a2608bc564
Status: Downloaded newer image for nginxdemos/hello:latest
reverseproxy_traefik_whoami_1 is up-to-date
reverseproxy_traefik_traefik_1 is up-to-date
Creating reverseproxy_traefik_nginx_1 ... done
```

- Verify on the dashboard all the services running on docker using traefik as router/load balancer



- Check the second service is up and running on the host specified path at docker-compose file (localhost/service2)



So finally we have two different services running on a docker container exposed on traefik reverse proxy service.