**1. Using a bridge network (not the default one), create 2 container then test connection using “curl”**

docker network ls # list network

docker network create --driver bridge alpine-network # create user define bridge network

docker network inspect alpine-network # inspect alpine-network

docker run -dit --name alpine1 --network alpine-network alpine ash # run container alpine1 with alpine-network

docker run -dit --name alpine2 --network alpine-network alpine ash

docker network inspect alpine-network # inspec alpine-network

docker attach alpine1 # attach into container1

-> apk add curl

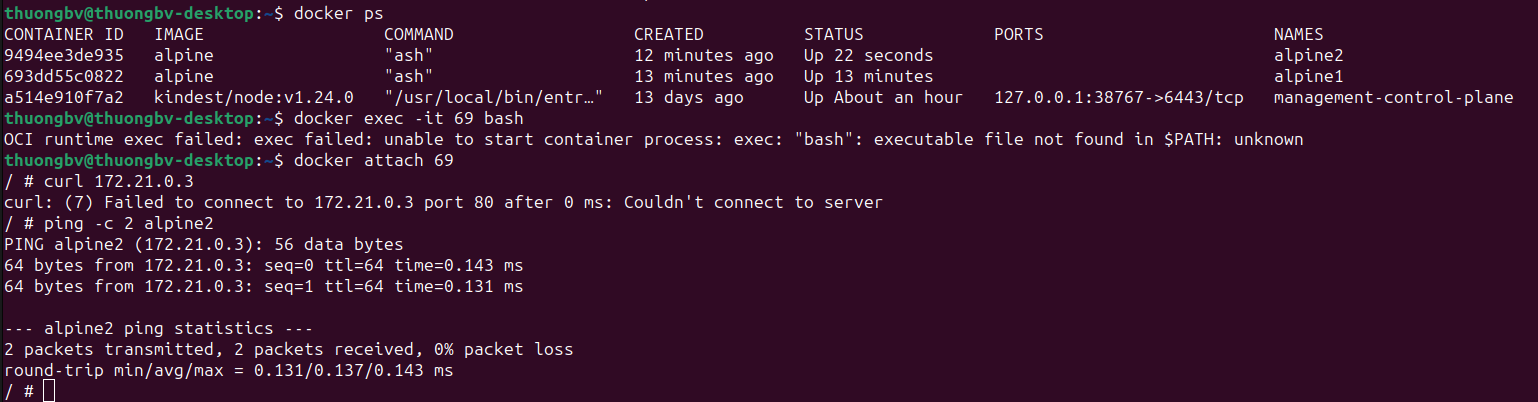
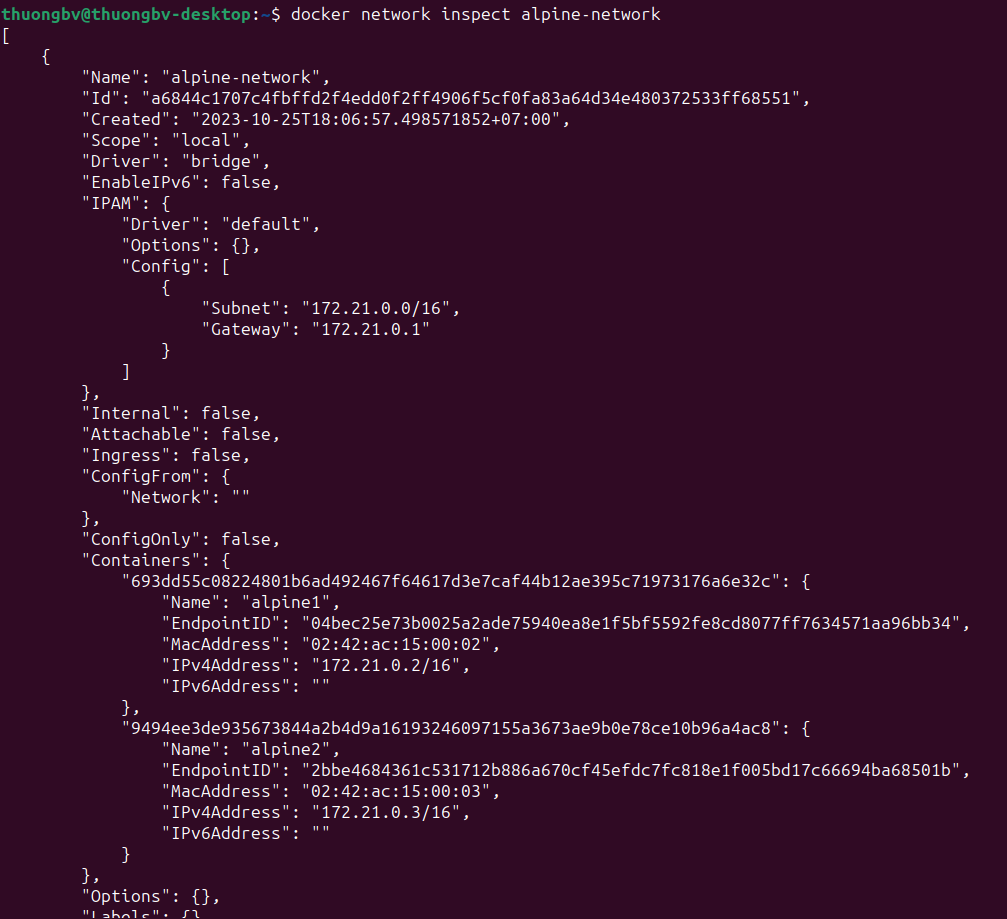
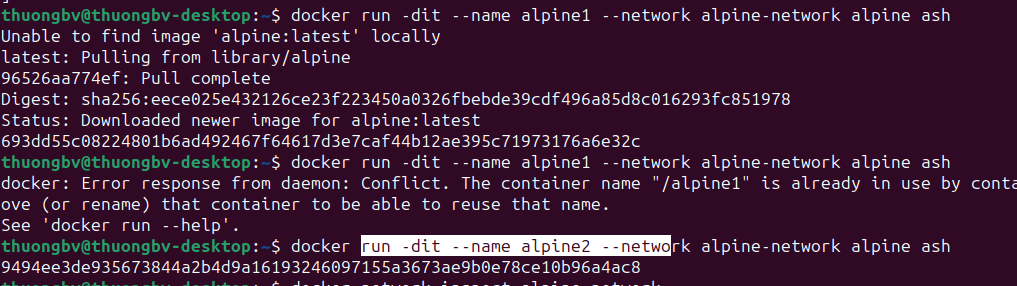
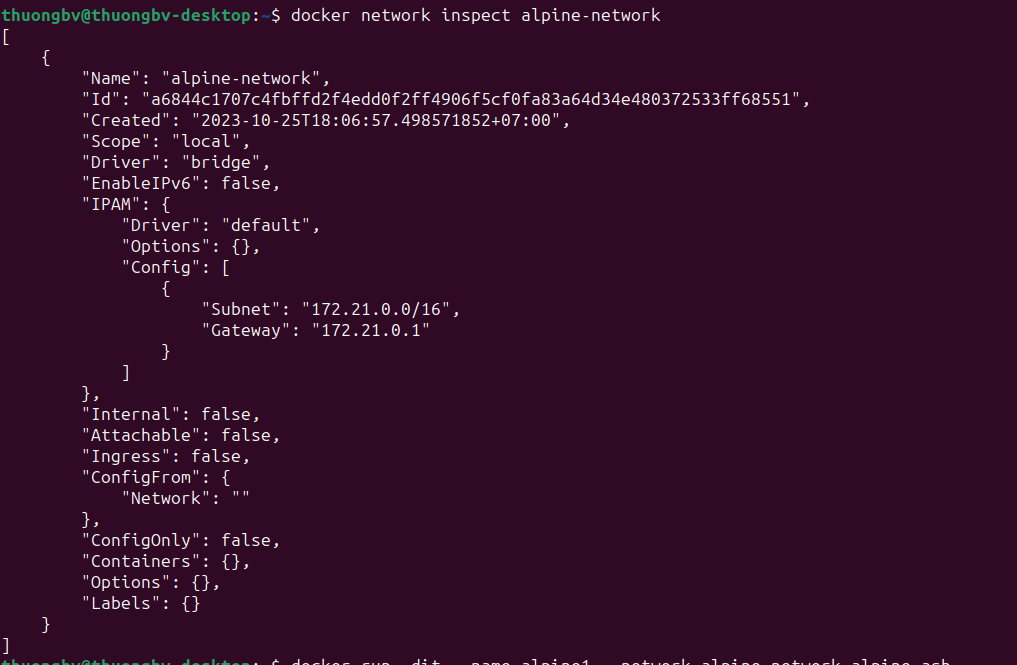
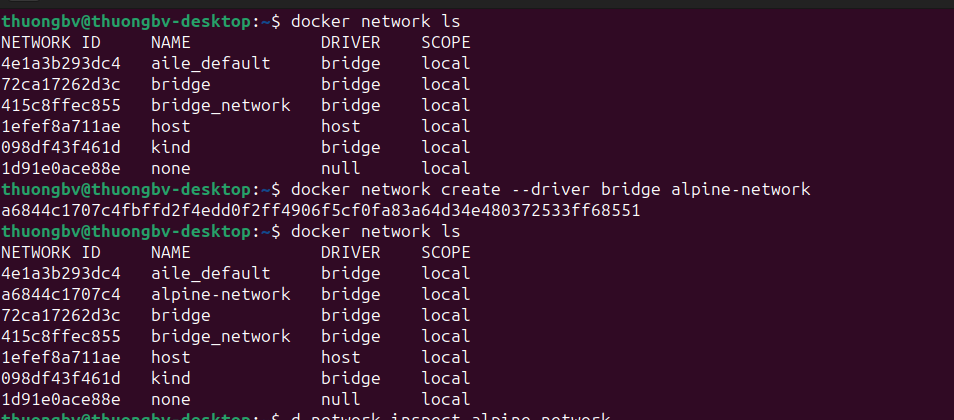
-> curl ip\_container1

-> ping -c 2 alpine2

**# note**

- sử dụng ping thành công vì bridge network đã phân giải DNS name thành địa chỉ ip

- curl không thành công có thể do tường lửa, hoặc cấu hình mạng không đúng

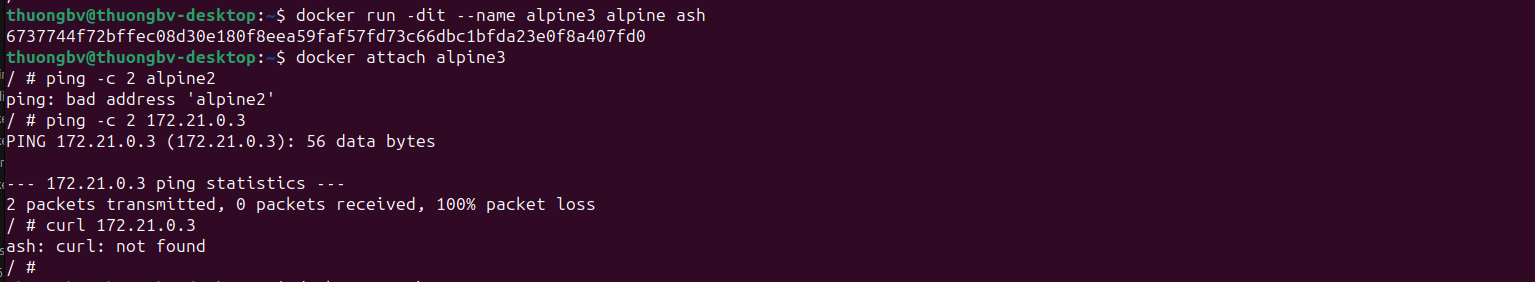


**2.Using 2 bridge networks, run a container on each network then test connection**

docker run -dit --name alpine3 alpine ash # run container3

docker attach alpine3 # attach into container3

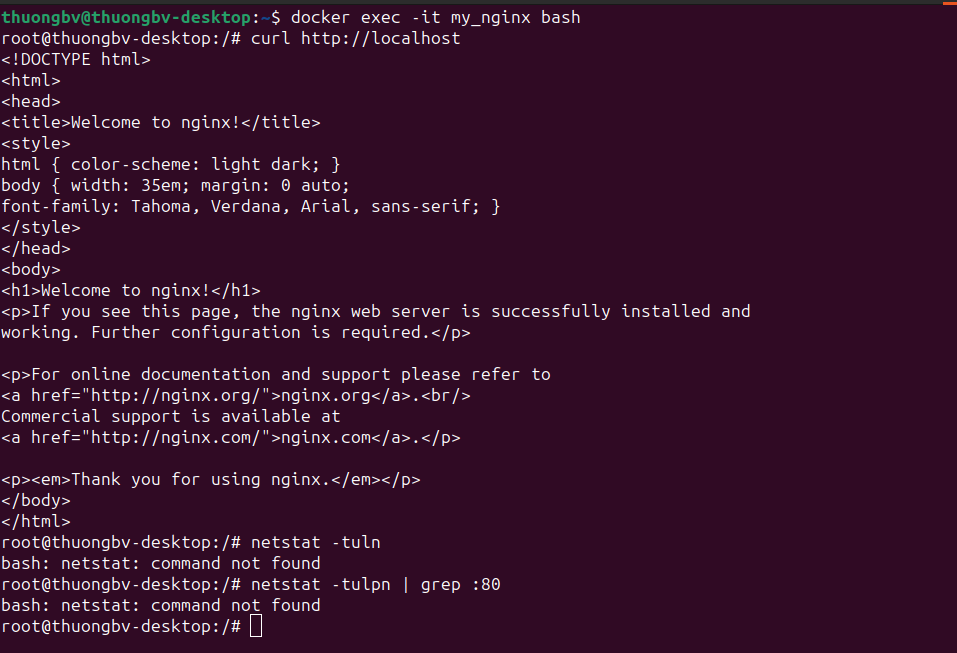
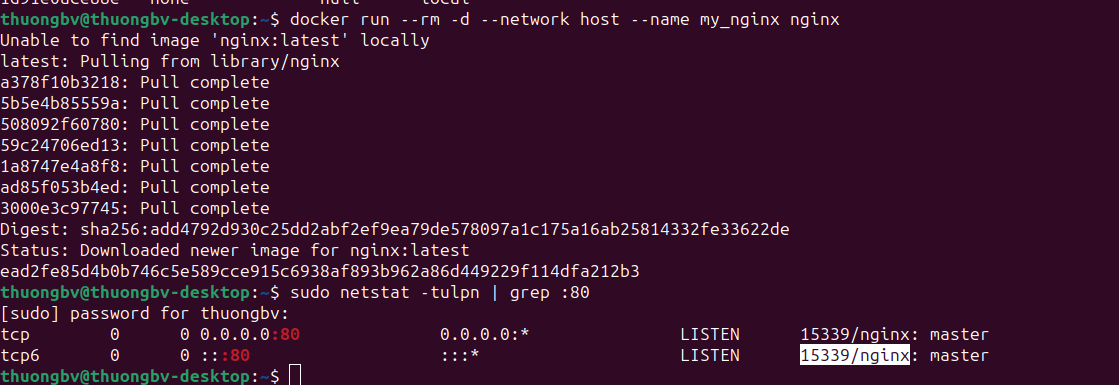
-> ping -c 2 alpine2 # fail vì không chung bridge network



**3. Create a host network, create a container running nginx, test connection using “curl” then verify which process is bound to port 80.**

docker run --rm -d --network host --name my\_nginx nginx # create nginx container with default host network

sudo netstat -tulpn | grep :80 # check port 80



docker exec -it my\_nginx bash # exec to container nginx

-> curl http://localhost # call to localhost

**4.Create a Macvlan, run a container with that network, “inspect” the container find its MacAddress then exec into the container and verify using “ip a”.**

docker network create -d macvlan --subnet=192.168.1.0/24 --gateway=192.168.1.1 -o parent=eth0.10 my\_macvlan\_network # Create macvlan network

-> Error response from daemon: -o parent interface was not found on the host: eth0

docker run -d --name my\_macvlan\_container --network my\_macvlan\_network alpine ash # create container with macvlan network

docker inspect my\_macvlan\_network

docker attach my\_macvlan\_container

-> ip a

**5. Do the same as above but with 802.1q trunked brigde mode**

docker network create -d macvlan --subnet=172.16.86.0/24 --gateway=172.16.86.1 -o parent=eth0.10 my-8021q-macvlan-net

docker run --rm -itd --network my-8021q-macvlan-net --name my\_macvlan\_container2 alpine:latest ash

docker container inspect my\_macvlan\_container2

PS:   
- em bị lỗi không tạo được Macvlan network, em có thử search thì thấy do phiên bản docker. Em có note lại quy trình làm ở trên rồi ạ.   
- Kiểu chữ này là do em copy từ vscode rồi paste vào WPS writer mà chưa biết cách config lại ạ. em có để định dạng docx để anh format lại cho dễ đọc