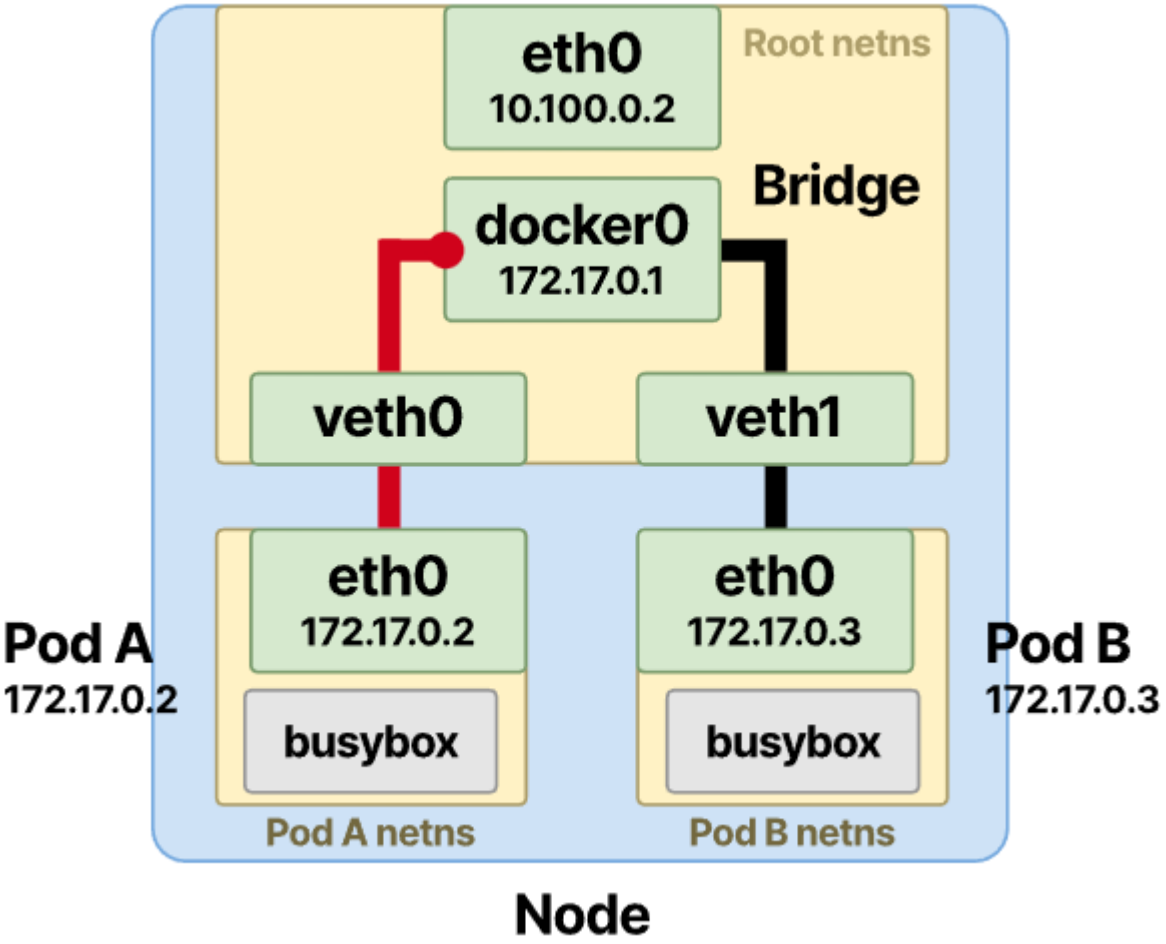
도커 네트워크

도커 네트워크 개요

- 도커 컨테이너 및 서비스는 도커 네트워크를 통해 격리된 컨테이너 간의 네트워크 연결뿐만 아니라 도커 외의 다른 애플리케이션 워크로드와도 연결이 가능
- 도커 네트워크의 하위 시스템 연결을 위해 도커 네트워크 드라이버를 사용하여 상호 간 통신이 가능해진다.

도커 네트워크 정의

- 도커 설치 시 기본적으로 제공되는 docker0는 소프트웨어적으로 구현된 가상 이더넷 브리지 네트워크이고, 이것을 격리된 컨테이너들의 상호 간 통신을 제공
- 별도의 브리지 네트워크를 생성하여 연결값으로 설정하지 않는 한 실행되는 모든 컨테이너는 docker0 브리지에 연결되어 172.17.0.0/16의 CIDR 범위로 IP 주소가 할당된다. /16은 최대 65,536개의 IP 주소 범위를 가진다.
- 도커 브리지 네트워크 연결



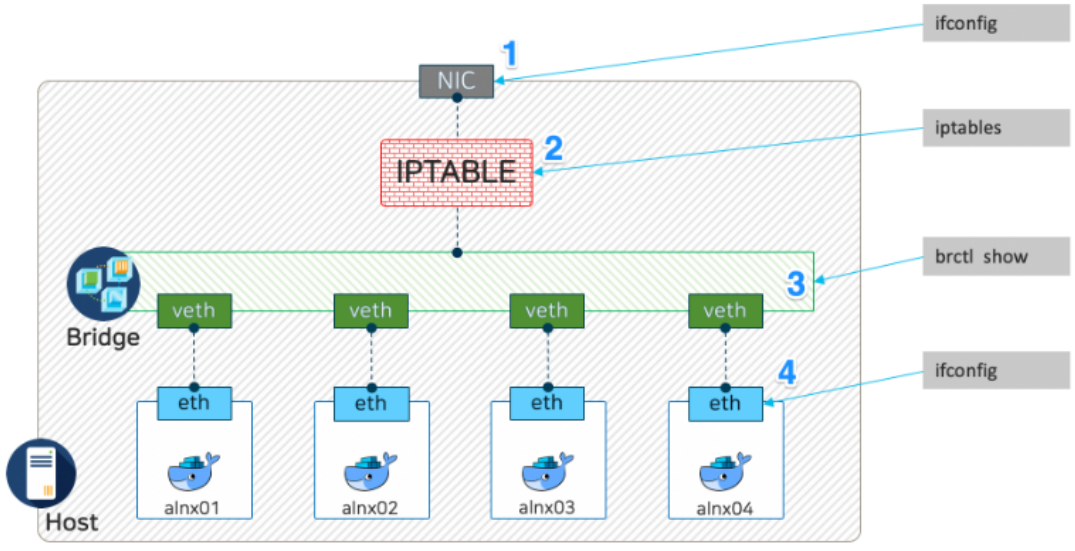
4가지 네트워크 인터페이스

- enp0s8: 우분투 리눅스의 네트워크 카드
- docker0: 도커 설치 시 기본적으로 제공되는 브리지 네트워크로 172.17.0.1 주소를 갖는다.
    - docker0 브리지는 소프트웨어적인 스위치 방식으로 동작하며, 일반적인 스위치 방식와는 다르게 DHCP로 연결된 컨테이너에 사전에 정의된 IP 풀을 할당한다.
- vethxxxxxxx: OSI 7계층 서비스 모델의 2계층 서비스로 컨테이너 내부에 제공되는 네트워크 인터페이스 eth0와 한 쌍으루 제공되어 docker0와 가상의 터널링 네트워크를 제공한다.
- eth0: 도커 컨테이너에 생성되는 기본 네트워크 인터페이스명으로 docker0를 게이트웨이로 사용한다. 그림에서는 172.16.0.2 ~ 3을 실행되는 컨테이너에 자동 할당하고 있다.

```
# 호스트 운영체제에 기본으로 설치된 docker0의 IP 주소를 확인
$ ifconfig docker0
```

```
# 현재 설정되어 있는 도커 네트워크 드라이버 방식을 조회
$ docker network ls
$ docker run -it -d --name ctn1 ubuntu:14.04
$ dockdf run -it -d --name ctn2 ubuntu:14.04
$ docker ps
$ docker inspect ctn1 | grep IPAddress
$ docker inspect ctn2 | grep IPAddress
$ docker exec ctn1 ifconfig
$ dock er exec ctn1 route
$ docker exec ctn1 ip addr
# 컨테이너 수만큼의 가상 네트워크 인터페이스 vthxxxxxxx가 자동 추가되었다
IP 주소가 할당되지 않는 터널링 서비스만 제공
$ ifconfig
# 브리지 네트워크 인터페이스 조회 도구인 brctl을 사용
$ brctl show
$ sudo apt install bridge-utils
$ brctl show
```

- 도커 네트워크 인터페이스 정보 조회




```
$ docker info | grep Network
Network: bridge host ipvlan macvlan null overlay
```

- 도커 네트워크 드라이버는 도커 엔진의 일부이며 추가 구성할 필요는 없다. docker run 사용 시 - -net(--network) 옵션을 이용해 선택할 수 있고, docker network 명령을 통해 호출하여 사용
    - bridge: 기본 네트워크 드라이버로 컨테이너 실행 시 별도의 네트워크 지정 없이 독립적으로 실행되는 애플리케이션 컨테이너를 실행하는 경우 사용. 단 브리지 모드는 동일 호스트 상의 도커 컨테이너만 적용된다.
    - host: 컨테이너의 호스트 모드를 사용하면 컨테이너와 호스트 간의 네트워크 격리를 제고하고 호스트의 네트워킹을 직접 사용할 수 있다. 이 기능을 통해 컨테이너 애플리케이션에 별도의 포트 연결(-p 호스트포트:컨테이너포트) 없이 호스트의 포트르를 이용하여 바로 서비스할 수 있다.
    - overlay: 다중 호스트 도커 서버를 이용한 클러스터(도커 스웜) 등을 이용할 경우 도커 데몬 간의 연결을 통해 컨테이너 서비스를 수행할 수 있다. 이 옵션을 사용하면 컨테이너 간에 운영체제 수준의 라우팅을 사용하지 않아도 된다. 도커 클러스터인 도커 스웜 구축 시 호스트와 호스트 간의 컨테이너 연결에 사용된다.
    - none: 컨테이너의 네트워크를 사용하지 않도록 설정한다. none 네트워크로 설정을 하면 네트워크 인터페이스는 lo 인터페이스만 존재한다. 컨테이너가 호스트 네트워킹 스택에서 완전히 분리되는 것으로 컨테이너는 외부와의 통신이 단절된다.
    - 컨테이너 네트워크: container:공유받을컨테이너이름 옵션은 컨테이너의 네트워크 네임스페이스 스택을 (IP 주소, Mac 주소 등)을 공유하여 사용할 수 있게 한다.
    - 사용자 정의 네트워크: docker network create 명령을 통해 사용자가 직접 생성한 도커 네트워크로 아무런 옵션을 주지 않고 생성하면 docker0 IP 대역의 다른 CIDR을 지정하여 생성된다.