

```
faofernando@war-machine:~$ docker port web1
80/tcp -> 0.0.0.0:5001
80/tcp -> :::5001
```

Porta do host e do container

```
faofernando@war-machine:~$ docker run -d -p 5002:80/udp --name=web2 my-apache
584cb3486405691dfce63d777e18aca3b4839d835c976567d249517400583e46
faofernando@war-machine:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
584cb3486405   my-apache  "apache2ctl -D FOREG..." 3 seconds ago  Up 2 seconds  80/tcp, 0.0.0.0:5002->80/udp, :::5002->80/udp  web2
d98731893eae   my-apache  "apache2ctl -D FOREG..." 2 days ago    Up 2 minutes  0.0.0.0:5001->80/tcp, :::5001->80/tcp  web1
faofernando@war-machine:~$ docker port web2
80/udp -> 0.0.0.0:5002
80/udp -> :::5002
```

Iniciando um container c/ uma conexao udp porta 5002

```
root@leandrocsi:/home/leandro# ip -f inet a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 68324sec preferred_lft 68324sec
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    inet 192.168.56.106/24 brd 192.168.56.255 scope global dynamic noprefixroute enp0s8
        valid_lft 1115sec preferred_lft 1115sec
4: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
root@leandrocsi:/home/leandro#
```

ip -f inet a = Para sabermos quais endereços ips estão disponíveis para o docker host é necessário digitar esse comando e analisar as interfaces 2 e 3. Não é necessário analisar o LOOPBACK e nem o Docker 0 . O interessante é apenas as interfaces.

Procedimento para definir um ip no container :

```
docker run -d -p 192.168.56.106:5003:80 --name=web3 my-apache
```

```
root@leandrocsi:/home/leandro# ip -f inet a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 68324sec preferred_lft 68324sec
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    inet 192.168.56.106/24 brd 192.168.56.255 scope global dynamic noprefixroute enp0s8
        valid_lft 1115sec preferred_lft 1115sec
4: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
root@leandrocsi:/home/leandro# docker run -d -p 192.168.56.106:5003:80 --name=web3 my-apache
7c1ec66fd0511b23faa5fbd5e4e85679f4a4d114e6c9e6c91bda90189336435
root@leandrocsi:/home/leandro# docker port web3
80/tcp -> 192.168.56.106:5003
root@leandrocsi:/home/leandro#
```

Podemos usar a -P que irá definir todas as portas expostas no dockerfile será atribuída a portas aleatórias no dockerhost

```
docker build -t="too-many-ports" .
```

```
faofernando@war-machine:~/Projetos/docker/10 - Redes no Docker/05 - Visualizando Portas Expostas$ docker build -t="too-many-ports" .
Sending build context to Docker daemon 2.048kB
Step 1/5 : FROM ubuntu:18.04
----> 8d5df41c547b
Step 2/5 : RUN apt-get update && apt-get install -y iproute2 iputils-ping traceroute apache2
----> Using cache
----> 6a416e540a15
Step 3/5 : EXPOSE 80 500 600 700 800 900
----> Running in 92547d67cada
Removing intermediate container 92547d67cada
----> 3f0578b58bc1
Step 4/5 : ENTRYPOINT ["apache2ctl"]
----> Running in 2ee09b3de0da
Removing intermediate container 2ee09b3de0da
----> d2dbc576870e
Step 5/5 : CMD ["-D", "FOREGROUND"]
----> Running in f19a6ad64ce8
Removing intermediate container f19a6ad64ce8
----> 53b757a8a96c
Successfully built 53b757a8a96c
Successfully tagged too-many-ports:latest
```

```
faofernando@war-machine:~/Projetos/docker/10 - Redes no Docker/05 - Visualizando Portas Expostas$ docker port too-many-ports
500/tcp -> 0.0.0.0:49157
500/tcp -> :::49157
600/tcp -> 0.0.0.0:49156
600/tcp -> :::49156
700/tcp -> 0.0.0.0:49155
700/tcp -> :::49155
80/tcp -> 0.0.0.0:49158
80/tcp -> :::49158
800/tcp -> 0.0.0.0:49154
800/tcp -> :::49154
900/tcp -> 0.0.0.0:49153
900/tcp -> :::49153
```

Todas as portas do container expostas em portas aleatórias no docker host