

NAME

`docker-network-connect` - Connect a container to a network

SYNOPSIS

`docker network connect [OPTIONS] NETWORK CONTAINER`

DESCRIPTION

Connects a container to a network. You can connect a container by name or by ID. Once connected, the container can communicate with other containers in the same network.

```
$ docker network connect multi-host-network container1
```

You can also use the `docker run --network=<network-name>` option to start a container and immediately connect it to a network.

```
$ docker run -itd --network=multi-host-network --ip 172.20.88.22 --ip6 2001:db8::8822 busybox
```

You can pause, restart, and stop containers that are connected to a network. A container connects to its configured networks when it runs.

If specified, the container's IP address(es) is reapplied when a stopped container is restarted. If the IP address is no longer available, the container fails to start. One way to guarantee that the IP address is available is to specify an `--ip-range` when creating the network, and choose the static IP address(es) from outside that range. This ensures that the IP address is not given to another container while this container is not on the network.

```
$ docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 multi-host-network
```

```
$ docker network connect --ip 172.20.128.2 multi-host-network container2
```

To verify the container is connected, use the `docker network inspect` command. Use `docker network disconnect` to remove a container from the network.

Once connected in network, containers can communicate using only another container's IP address or name. For `overlay` networks or custom plugins that support multi-host connectivity, containers connected to the same multi-host network but launched from different Engines can also communicate in this way.

You can connect a container to one or more networks. The networks need not be the same type. For example, you can connect a single container bridge and overlay networks.

OPTIONS

- alias=[]** Add network-scoped alias for the container

- driver-opt=[]** driver options for the network

- h, --help[=false]** help for connect

- ip=""** IPv4 address (e.g., 172.30.100.104)

- ip6=""** IPv6 address (e.g., 2001:db8::33)

- link=** Add link to another container

- link-local-ip=[]** Add a link-local address for the container

SEE ALSO**docker-network(1)**