Exercise Worksheet

[www.vomtom.at](http://www.vomtom.at)

# From the Course: Understanding Docker Run, Dockerfile, Docker-Compose for Beginners

## Understanding the Basic Docker Bridge and Host Network

By default, docker creates an overlay network over all containers. A bridge. It’s called “docker0”.

From the previous lectures there should be some networks left:

docker network ls

* Brings up a list of all networks

docker network prune

* Should delete all networks
* Just to clean up – you don’t *have* to do this, but sometimes it’s good to start fresh

docker network ls

* Should now just list the necessary networks for docker to function
* A bridge
* A host
* A null

docker run --rm --name my-webserver -d httpd

* Start an apache webserver (httpd container)

docker inspect my-webserver

* Check the IP address of the container

Open <http://172.17.0.2> (or the IP address of your container)

* It won’t let you connect to it
* Unless you forward a port to your host with -p 8080:80 or so…

docker run --rm tomw1808/mycurl my-webserver

* Should download and run an image called “tomw1808/curl” which is just an ubuntu alpine with curl installed
* And curl “my-webserver”
  + Basically, the same as executing “curl my-webserver” on any Linux
* It will end in an error

docker run --rm tomw1808/mycurl 172.17.0.2

* Will output you the HTML of the Webserver
* “It Works!”

docker stop my-webserver

* Stops the webserver container

docker network create simple-network

* Creates a new bridge network called “simple-network”

docker run --rm -d --name my-webserver --network simple-network httpd

* Start the webserver again attaching it to the “simple-network” we created earlier

docker run --rm --network simple-network appropriate/curl my-webserver

* Now the name binding works
* “It Works!”

docker inspect my-webserver

* Get the IP Address of your webserver
* It should be 172.22.0.2 (or so – copy the IP of your container here)

docker run --rm tomw1808/mycurl 172.22.0.2

* Run curl without the network, on the docker0 network
* You won’t be able to connect to the webserver
* It’s segregated from the other network
* Ctrl-c to stop

docker stop my-webserver

* Stop the container again

docker network rm simple-network

* Cleanup: remove the simple-network again