Single Host Networking

By default, any Docker container or host will get an *IP* address that will give it the possibility to communicate with other containers in the same host or with the host machine.

Containers in a single host, could also communicate and reach the outside world.

Create a simple container:

docker run -it -d --name my_container busybox

And test if you can ping Google:

```
docker exec -it my_container ping -w3 google.com
PING google.com (216.58.204.142): 56 data bytes 64 bytes from
216.58.204.142: seq=1 ttl=48 time=2.811 ms
--- google.com ping statistics ---
3 packets transmitted, 1 packets received, 66% packet loss
round-trip min/avg/max = 2.811/2.811/2.811 ms
```

Now if you inspect the container using `docker inspect my_container` you will be able to see its network configuration and its *IP* address:

```
"NetworkSettings": {
            "Bridge": "",
            "SandboxID":
"555a60eaffdb4b740f7b869bac61859ecca1e39be95ee5856ca28019509e4
255",
            "HairpinMode": false,
            "LinkLocalIPv6Address": "",
            "LinkLocalIPv6PrefixLen": 0,
            "Ports": {},
            "SandboxKey":
"/var/run/docker/netns/555a60eaffdb",
            "SecondaryIPAddresses": null,
```

```
            "SecondaryIPv6Addresses": null,
            "EndpointID":
"20b1b218462e6771155de75788f53b731bbff12019d977aefa7094f572758
87d",
            "Gateway": "172.17.0.1",
            "GlobalIPv6Address": "",
            "GlobalIPv6PrefixLen": 0,
            "IPAddress": "172.17.0.2",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
            "MacAddress": "02:42:ac:11:00:02",
            "Networks": {
                "bridge": {
                    "IPAMConfig": null,
                    "Links": null,
                    "Aliases": null,
                    "NetworkID":
"2094b393faacbb1cc049f1f136437b1cce6fc41abc304cf2c1ae558a62c5e
e2e",
                    "EndpointID":
"20b1b218462e6771155de75788f53b731bbff12019d977aefa7094f572758
87d",
                    "Gateway": "172.17.0.1",
                    "IPAddress": "172.17.0.2",
                    "IPPrefixLen": 16,
                    "IPv6Gateway": "",
                    "GlobalIPv6Address": "",
                    "GlobalIPv6PrefixLen": 0,
                    "MacAddress": "02:42:ac:11:00:02"
                }
            }
        }
```


my_container has the *IP* address *172.17.0.2* that the host could reach:

```
ping -w1 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.050 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.045 ms
--- 172.17.0.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.045/0.047/0.050/0.007 ms
```

If you run a web server, your users must reach the port 80 (or 443) of your server, in this case an *nginx* container, for example,

should be reached at its port 80 (or 443) and it is done through port forwarding that connects it to the host machine and then an external network (Internet in our case).

Let's create the web server container, forward the port host port 8080 to the container port 80 and test how it responds:

docker run -d -p 8080:80 --name my_web_server nginx

Ngninx should reply if your port 8080 is not used by other applications:

```
curl http://0.0.0.0:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully
installed and
working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

In a single host, containers are able to see each other, to see the external world (if they are not running in isolated networks) and they can receive traffic from an external network.