

A abordagem dessa aula pode ser complementada com a documentação do Docker Daemon. Nos links a seguir você pode ver mais detalhes sobre segurança e sobre como expor seu Docker Engine ou Daemon via rede sem grandes riscos.

<https://docs.docker.com/v17.09/engine/admin/#configure-the-docker-daemon>

<https://docs.docker.com/engine/reference/commandline/dockerd/>

```
File Edit View Search Terminal Help
root@leandrocsi:/home/leandro# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:9a:f9:fc brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 85694sec preferred_lft 85694sec
    inet6 fe80::d640:5431:b5a9:7117/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:cd:c7:b2 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.106/24 brd 192.168.56.255 scope global dynamic noprefixroute enp0s8
        valid_lft 1002sec preferred_lft 1002sec
    inet6 fe80::8183:ad2:d445:b126/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:61:5a:ec:e6 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:61ff:fe5a:ece6/64 scope link
        valid_lft forever preferred_lft forever
```

Figura 1: Exibe o ip

```
faofernando@war-machine:~$ service docker stop
Warning: Stopping docker.service, but it can still be activated by:
docker.socket
```

Figura 2: Pára o service do docker

```
root@leandrocsi:/home/leandro# dockerd -H tcp://192.168.56.106:2375 &
[1] 3391
root@leandrocsi:/home/leandro# WARN[2019-10-22T22:12:23.340953117-03:00] [!] DON'T BIND ON ANY IP ADDRESS WITHOUT setting --tlsverify IF YOU DON'T KNOW WHAT YOU'RE DOING [!]
INFO[2019-10-22T22:12:23.344344750-03:00] systemd-resolved is running, so using resolvconf: /run/systemd/resolve/resolv.conf
INFO[2019-10-22T22:12:23.344844045-03:00] parsed scheme: "unix" module=grpc
INFO[2019-10-22T22:12:23.344891200-03:00] scheme "unix" not registered, fallback to default scheme module=grpc
INFO[2019-10-22T22:12:23.344939574-03:00] parsed scheme: "unix" module=grpc
INFO[2019-10-22T22:12:23.344971769-03:00] scheme "unix" not registered, fallback to default scheme module=grpc
INFO[2019-10-22T22:12:23.345944210-03:00] [graphdriver] using prior storage driver: overlay2
INFO[2019-10-22T22:12:23.347082135-03:00] ccResolverWrapper: sending new addresses to cc: [{unix:///run/containerd/containerd.sock 0 <nil>}] module=grpc
INFO[2019-10-22T22:12:23.347129529-03:00] ClientConn switching balancer to "pick_first" module=grpc
INFO[2019-10-22T22:12:23.347182361-03:00] pickfirstBalancer: HandleSubConnStateChange: 0xc42015f140, CONNECTING module=grpc
INFO[2019-10-22T22:12:23.347448750-03:00] pickfirstBalancer: HandleSubConnStateChange: 0xc42015f140, READY module=grpc
INFO[2019-10-22T22:12:23.347497222-03:00] ccResolverWrapper: sending new addresses to cc: [{unix:///run/containerd/containerd.sock 0 <nil>}] module=grpc
INFO[2019-10-22T22:12:23.347530383-03:00] ClientConn switching balancer to "pick_first" module=grpc
INFO[2019-10-22T22:12:23.347576012-03:00] pickfirstBalancer: HandleSubConnStateChange: 0xc42015f3d0, CONNECTING module=grpc
INFO[2019-10-22T22:12:23.34767317-03:00] pickfirstBalancer: HandleSubConnStateChange: 0xc42015f3d0, READY module=grpc
WARN[2019-10-22T22:12:23.357471910-03:00] Your kernel does not support swap memory limit
WARN[2019-10-22T22:12:23.357559602-03:00] Your kernel does not support cgroup rt period
WARN[2019-10-22T22:12:23.357596479-03:00] Your kernel does not support cgroup rt runtime
WARN[2019-10-22T22:12:23.357627337-03:00] Your kernel does not support cgroup blkio weight
WARN[2019-10-22T22:12:23.357657725-03:00] Your kernel does not support cgroup blkio weight_device
INFO[2019-10-22T22:12:23.358025592-03:00] Loading containers: start.
INFO[2019-10-22T22:12:23.452306897-03:00] Default bridge (docker0) is assigned with an IP address 172.17.0.1/16. Daemon option --bip can be used to set a preferred IP address
INFO[2019-10-22T22:12:23.488896132-03:00] Loading containers: done.
WARN[2019-10-22T22:12:23.491788139-03:00] Failed to retrieve runc version: unknown output format: runc version spec: 1.0.1-dev
INFO[2019-10-22T22:12:23.526829773-03:00] Docker daemon
INFO[2019-10-22T22:12:23.529810030-03:00]
INFO[2019-10-22T22:12:23.579431142-03:00]
```

Figura 3: dockerd = dockerd daemon ; H = é para especifica o host ; &= Devolva o prompt de comando