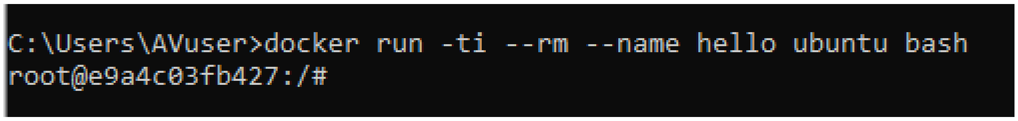
Process

A container is started with the main process called the INIT process. The main process can be divided into various sub-processes. If the init process is terminated, the container will be exited.

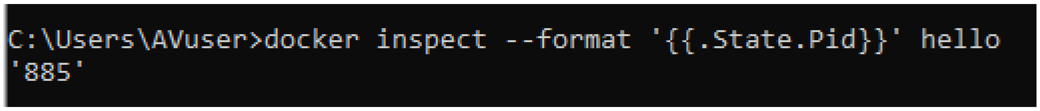
Terminating the INIT process of a conatiner

1. Create an Ubuntu container named hello with running a bash process. The tag –ti is used for interactive terminal, --rm for self cleanup, --name for providing the name for the container.



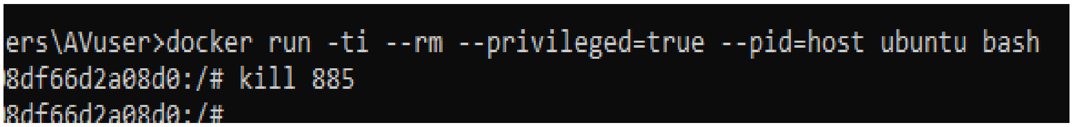
```
C:\Users\AVuser>docker run -ti --rm --name hello ubuntu bash
root@e9a4c03fb427:/#
```

2. To identify the Process ID of the hello container, use the command "**docker inspect**". The tag **"--format `{{.State.Pid}}`"** is specified to retreive only the process id of the container.



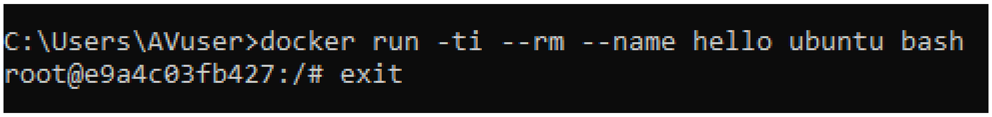
```
C:\Users\AVuser>docker inspect --format '{{.State.Pid}}' hello
'885'
```

3. Create a privileged Ubuntu container. The tag "--privileged=true" and "--pid=host" is used to turn off all the securities. Inside the container, use the command **"kill <process_id>"** to terminate the init process of the hello container.



```
ers\AVuser>docker run -ti --rm --privileged=true --pid=host ubuntu bash
8df66d2a08d0:/# kill 885
8df66d2a08d0:/#
```

4. Once the init process is terminated, the hello container will be exited with all its sub-processes terminating abruptly.



```
C:\Users\AVuser>docker run -ti --rm --name hello ubuntu bash
root@e9a4c03fb427:/# exit
```