Task 1

Let's check how to check docker version

**Try running below commands**  
docker version

***Hint* : to get the command output in page wise use below command**  
docker version | less  
**Run q to exit from the less**  
q

#### Q 10

##### Let's look into the file level isolation

**We need a directory for the chroot to test**  
chr=/home/testuser/testroot  
**Let's create directory**  
mkdir -p $chr

**Now we will create some directory to hold few binaries and library from system**  
mkdir -p $chr/{bin,lib,lib64}  
**Let's get into our chroot directory**  
cd $chr

**Now let's copy one binary that we can provide as minimalist Linux environment**  
cp -v /bin/bash $chr/bin

**Now let's check the dependancies for the binaries and copy them to our directory**  
ldd /bin/bash  
**Let's save it in variable for simplicity**  
list="$(ldd /bin/bash | egrep -o '/lib.\*\.[0-9]')"  
**Let's check the value of variable**  
echo $list  
**Let's use for loop to copy content**  
for i in $list; do cp -v --parents "$i" "${chr}"; done

**SO Finally! we are ready to chroot**  
sudo chroot $chr bin/bash

**Let's try to run ls command**  
echo 'This is testing of the chroot'  
Ignore warning relate to setLocale

**Let's check if we are able to run other commands**  
touch test.txt

**Let's get out of the JAIL**  
exit

**Let's check Again! if we are able to run other commands**  
touch test.txt

ubuntu $ docker version

Client:

Version: 20.10.12

API version: 1.41

Go version: go1.16.2

Git commit: 20.10.12-0ubuntu2~20.04.1

Built: Wed Apr 6 02:14:38 2022

OS/Arch: linux/amd64

Context: default

Experimental: true

Server:

Engine:

Version: 20.10.12

API version: 1.41 (minimum version 1.12)

Go version: go1.16.2

Git commit: 20.10.12-0ubuntu2~20.04.1

Built: Thu Feb 10 15:03:35 2022

OS/Arch: linux/amd64

Experimental: false

containerd:

Version: 1.5.9-0ubuntu1~20.04.4

GitCommit:

runc:

Version: 1.1.0-0ubuntu1~20.04.1

GitCommit:

docker-init:

Version: 0.19.0

GitCommit:

ubuntu $ docker version | less

ubuntu $

ubuntu $ echo Let me setup scenario

Let me setup scenario

ubuntu $ while [ ! -f /tmp/background1 ]; do sleep 1; done

ubuntu $ echo DONE

DONE

ubuntu $ chr=/home/testuser/testroot

ubuntu $ mkdir -p $chr

ubuntu $ mkdir -p $chr/{bin,lib,lib64}

ubuntu $ cd $chr

ubuntu $ cp -v /bin/bash $chr/bin

'/bin/bash' -> '/home/testuser/testroot/bin/bash'

ubuntu $ ldd /bin/bash

linux-vdso.so.1 (0x00007ffdbbafe000)

libtinfo.so.6 => /lib/x86\_64-linux-gnu/libtinfo.so.6 (0x00007f47b32a6000)

libdl.so.2 => /lib/x86\_64-linux-gnu/libdl.so.2 (0x00007f47b32a0000)

libc.so.6 => /lib/x86\_64-linux-gnu/libc.so.6 (0x00007f47b30ae000)

/lib64/ld-linux-x86-64.so.2 (0x00007f47b340c000)

ubuntu $ list="$(ldd /bin/bash | egrep -o '/lib.\*\.[0-9]')"

ubuntu $ echo $list

/lib/x86\_64-linux-gnu/libtinfo.so.6 /lib/x86\_64-linux-gnu/libdl.so.2 /lib/x86\_64-linux-gnu/libc.so.6 /lib64/ld-linux-x86-64.so.2

ubuntu $ for i in $list; do cp -v --parents "$i" "${chr}"; done

/lib/x86\_64-linux-gnu -> /home/testuser/testroot/lib/x86\_64-linux-gnu

'/lib/x86\_64-linux-gnu/libtinfo.so.6' -> '/home/testuser/testroot/lib/x86\_64-linux-gnu/libtinfo.so.6'

'/lib/x86\_64-linux-gnu/libdl.so.2' -> '/home/testuser/testroot/lib/x86\_64-linux-gnu/libdl.so.2'

'/lib/x86\_64-linux-gnu/libc.so.6' -> '/home/testuser/testroot/lib/x86\_64-linux-gnu/libc.so.6'

'/lib64/ld-linux-x86-64.so.2' -> '/home/testuser/testroot/lib64/ld-linux-x86-64.so.2'

ubuntu $ echo 'This is testing of the chroot'

This is testing of the chroot

ubuntu $ touch test.txt

ubuntu $ touch test.txt

ubuntu $