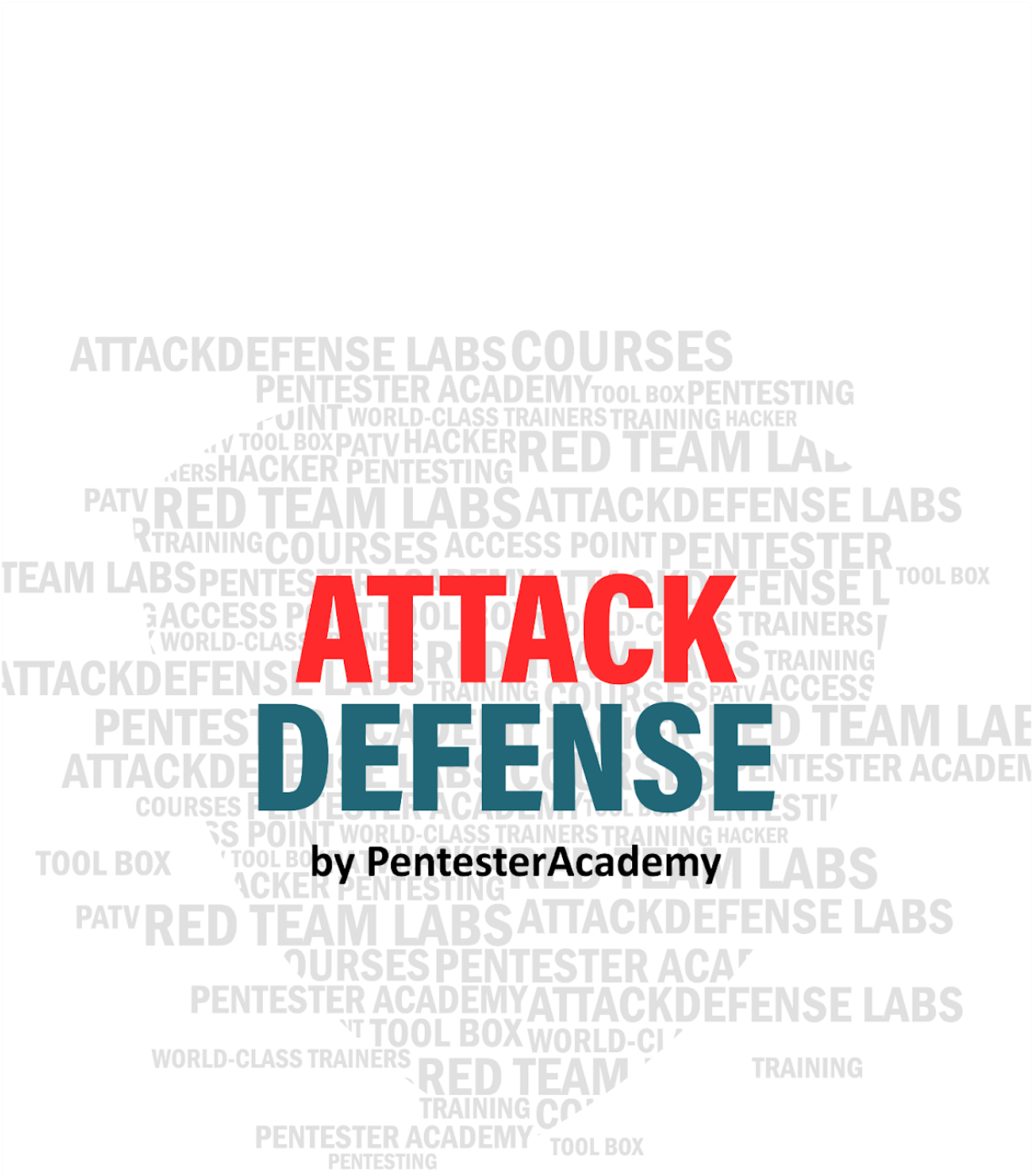
ATTACK
DEFENSE
by PentesterAcademy

Name	Containers With Runc
URL	https://attackdefense.com/challengedetails?cid=1462
Type	Docker Security : Container Basics

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: Fetch the image and run it using runc!

Solution:

Step 1: Check the list of repos on private docker repository

Command: curl --insecure https://registry:5000/v2/_catalog

```
root@localhost:~# curl --insecure https://registry:5000/v2/_catalog
{"repositories":["alpine","memcached","nginx","ubuntu"]}
root@localhost:~#
```

There are four repositories.

Step 2: Check the tags for “alpine” repository.

Command: curl --insecure https://registry:5000/v2/alpine/tags/list

```
root@localhost:~# curl --insecure https://registry:5000/v2/alpine/tags/list
{"name":"alpine","tags":["3.7","latest"]}
root@localhost:~#
```

Step 3: Pull “alpine:latest” docker image using skopeo and save it in OCI format.

Commands: skopeo --insecure-policy copy --src-tls-verify=false docker://registry:5000/alpine oci:alpine-oci:latest

```
root@localhost:~# skopeo --insecure-policy copy --src-tls-verify=false docker://registry:5000/alpine oci:alpine-oci:latest
Getting image source signatures
Copying blob 89d9c30c1d48 done
Copying blob 4dc2274c3171 done
Copying config 9468df872c done
Writing manifest to image destination
Storing signatures
root@localhost:~#
```

Step 4: Check the image stored in OCI format.

Commands:
ls -l
find alpine-oci

```
root@localhost:~# ls -l
total 8
drwxr-xr-x 3 root root 4096 Dec  3 15:35 alpine-oci
drwxr-xr-x 3 root root 4096 Dec  3 15:35 snap
root@localhost:~#
```

```
root@localhost:~# find alpine-oci/
alpine-oci/
alpine-oci/blobs
alpine-oci/blobs/sha256
alpine-oci/blobs/sha256/9468df872c50673435ff2192244e09691e79ab8022998b7c5c476f2cea8e411d
alpine-oci/blobs/sha256/89d9c30c1d48bac627e5c6cb0d1ed1eec28e7dbdfbcc04712e4c79c0f83faf17
alpine-oci/blobs/sha256/66fed32ecb7ee554fad994c907610011520c642879318c1e1353fde6bbacf23a
alpine-oci/blobs/sha256/4dc2274c3171fd1f1957b3fa2c03e2a5d4fadc5171bf03afb0d643392ad9e868
alpine-oci/oci-layout
alpine-oci/index.json
root@localhost:~#
```

Step 5: Use umoci to convert OCI image to image bundle.

Command: umoci unpack --image alpine-oci alpinefs

```
root@localhost:~# umoci unpack --image alpine-oci alpinefs
root@localhost:~#
root@localhost:~# ls -l
total 12
drwxr-xr-x 3 root root 4096 Dec  3 15:35 alpine-oci
drwx------ 3 root root 4096 Dec  3 15:38 alpinefs
drwxr-xr-x 3 root root 4096 Dec  3 15:35 snap
root@localhost:~#
```

Step 6: Use the image bundle to create and run container using runc

Command: runc run -b alpinefs ctrid

```
root@localhost:~# runc run -b alpinefs ctrid
/ # whoami
root
/ # pwd
/
/ # ps
PID   USER     TIME  COMMAND
    1 root      0:00 /bin/sh
    8 root      0:00 ps
/ #
```

The container is running and the user can execute commands in it using shell.

References:

1. Umoci (https://github.com/openSUSE/umoci)
2. Runc (https://github.com/opencontainers/runc)
3. Skopeo (https://github.com/containers/skopeo)