CIS

**CRIO**

**/var/run/containers/storage/overlay-containers/<cont-id> /userdata 🡪 config.json copy + resolv.conf + hostname, pidfile**

**/var/lib/containers/storage/overlay-containers/<cont-id>/userdata -> config.json state.json**

**\*config.json : annotations could be used for port mapping, privileged, capabilities, apparmor(hopefully)**

**/proc/{procid}/attr/selinux**

**/proc/{procid}/attr/apparmor**

**/proc/19557/net/tcp -🡪 port mappings**

getpcaps {procid}

<https://linux-audit.com/linux-capabilities-101/>

CIS 5.1

CIS 5.3

**5.4: privilege**

**"io.kubernetes.cri-o.PrivilegedRuntime": under annotations**

**CIS 5.7 : don't map privileged ports <1025**

**"io.kubernetes.cri-o.PortMappings": "[]", under annotations**

**CIS 5.8 : ports**

**"io.kubernetes.cri-o.PortMappings": "[]", under annotations**

**CIS 5.9: network host**

**"io.kubernetes.cri-o.HostNetwork": "false",**

**CIS 5.11 :**

**/sys/fs/cgroup/cpu/Burstable/pod\_{podid}/crio-{containerid}/cpu.share**

**CIS 5.10:**

**/sys/fs/cgroup/memory/Burstable/pod\_{podid}/crio-{continaerid}/memory.limit\_in\_bytes**

CIS 5.12 : readonly root

config -> "root": {“readonly”}

1. CIS 5.5: mounts

Config.json -> mounts

1. CIS 5.13 Bind incoming container traffic to a specific host interface

Cconfig.json

"io.kubernetes.cri-o.PortMappings" -> there should be an ip

CIS 5.17 : devices

config.json -> [“linux”] [“resources”] [“devices”]

CIS 5.21 : seccomp

**"io.kubernetes.cri-o.SeccompProfilePath": "",**

5.24 Confirm cgroup usage (Scored)

state.json 🡪 "io.kubernetes.cri-o.CgroupParent":

5.28 pids limit

/sys/fs/cgroup/pids/ **Burstable/pod\_{podid}/crio-{continaerid}**/pids.max

1. CIS 5.6 don't run ssh

Get child processes and cat /proc/pid/cmdline == ssh

CIS 5.2 – selinux --- **/proc/{procid}/attr/selinux**

**Docker**

Docker containers are stored under

/var/lib/docker/containers/{id}

Under that : logs, config.json and exc. We can use config.json to get instance information like name, args and so on.

**run/containerd/io.containerd.runtime.v1.linux/moby/<containerid>**

* **Pid , config.json, rootfs**

docker inspect -f '{{.State.Pid}}' <containerid> gives you process id

state.json -🡪 **/run/docker/runtime-runc/moby/{contid}**

**host config: /var/lib/docker/containers/8f75c189620330b5103f4a9b23befda8c7d64903393d78d2b16120cc28c5255a**

* CIS 5.5 do not mount : /proc/{procid}/mounts -🡪??????

1. CIS 5.1
2. CIS 5.3
3. CIS 5.4: privilege

Hostconfig :Privileged

1. CIS 5.5: mounts

Config.json -> mounts

1. CIS 5.7: Don't map ports that are privileged <1025

/var/lib/docker/containers/{contid}/hostconfig.json

“PortBindings”

1. CIS 5.8: check ports

Same with above

1. CIS 5.9: host network

Hostconfig 🡪 "NetworkMode": "default",

1. CIS 5.10: memory limit

/sys/fs/cgroup/memory/docker/{contid}/memory.limit\_in\_bytes

1. CIS 5.11: cpu share

/sys/fs/cgroup/cpu/docker/{contid}/cpu.shares

1. CIS 5.12 : readonly root

Hostconfig -> "ReadonlyRootfs": false,

1. CIS 5.13 Bind incoming container traffic to a specific host interface

/var/lib/docker/containers/{contid}/hostconfig.json

“PortBindings”

* Host ip must not be empty

1. CIS 5.17 : devices

config.json -> [“linux”] [“resources”] [“devices”]

1. 5.21 Do not disable default seccomp profile (Scored)

config.json -> seccomp

1. 5.24 Confirm cgroup usage (Scored)

hostconfig -> CgroupParent

1. 5.28 pids limit

/sys/fs/cgroup/pids/docker/{contid}/pids.max

1. CIS 5.6 don't run ssh

Get child processes and cat /proc/pid/cmdline == ssh

Containerd

/run/containerd/io.containerd.runtime.v2.task/default/<contid> : config and init.pid!

run container as follows: ctr run docker.io/library/busybox:latest 0606’

CIS 5.1

CIS 5.3

1. CIS 5.5: mounts

Config.json -> mounts

/run/containerd/runc/default/0623/ -> state.json

CIS 5.10: memory limit

"/sys/fs/cgroup/cpu/{}/memory.limit\_in\_bytes”

CIS 5.11: cpu share

"/sys/fs/cgroup/cpu/{}/cpu.shares"

CIS 5.12 : readonly rootfs

State.json -> [“config”] [“readonlyfs”]

CIS 5.17 : devices

config.json -> [“linux”] [“resources”] [“devices”]

CIS 5.21 : seccomp

State.json -> [“config”] [“seccomp”]

5.24 Confirm cgroup usage (Scored)

"cgroupsPath" under “linux” in config.json

5.28 pids limit

/sys/fs/cgroup/pids/docker/{contid}/pids.max