**Security:**

**Vulnerability assessments:**

Static analysis using clair:

1. docker pull ubuntu:12.04 (we will use ubuntu version 12.04 as it is old and we can find more vulnerabilities)
2. docker run -d -p 5432:5432 --name db arminc/clair-db:latest (to start a clair database)
3. docker run -d -p 6060:6060 --link db:postgres --name clair arminc/clair-local-scan:latest (to start clair local scan container)

**Docker bench security:**

1. docker run -itd alpine
2. <https://github.com/docker/docker-bench-security>

docker run --rm --net host --pid host --userns host --cap-add audit\_control \

-e DOCKER\_CONTENT\_TRUST=$DOCKER\_CONTENT\_TRUST \

-v /etc:/etc:ro \

-v /usr/bin/containerd:/usr/bin/containerd:ro \

-v /usr/bin/runc:/usr/bin/runc:ro \

-v /usr/lib/systemd:/usr/lib/systemd:ro \

-v /var/lib:/var/lib:ro \

-v /var/run/docker.sock:/var/run/docker.sock:ro \

--label docker\_bench\_security \

docker/docker-bench-security

**Defences:**

**Using apparmor profiles:**

#include <tunables/global>

profile docker-nginx flags=(attach\_disconnected,mediate\_deleted) {

#include <abstractions/base>

file,

network,

capability,

deny /tmp/\*\* w,

deny /etc/passwd rwklx,

}

docker run -itd --security-opt apparmor=apparmor-profile alpine (run a container using apparmor-profile)

get a shell using exec and verify whether the rules are enforced or not.

touch /tmp/file (to create a file inside tmp)

**Using seccomp:**

touch /tmp/testfile

chmod 400 /tmp/testfile

**Make root user less powerful:**