Docker image building

Dockerfile

Why to build custom images

- To containerize your application with controlled source code
- Define your own libraries and packages
- To impose your source code
- Many other reason

Container image building tools

- Dockerfile
- Buildah
- Kaniko
- packer

Dockerfile

- A file based instruction to create image
- Dockerfile is having its own set of keywords
- Keywords :

  I) FROM , LABEL , ENV , ARG , RUN , WORKDIR , CMD , ADD

  ii) COPY , EXPOSE , ENTRYPOINT ,STOPSIGNAL,ONBUILD

  iii) USER, MAINTAINER ,HEALTHCHECK,SHELL

Sample dockerfile

```
FROM ubuntu

LABEL name=ashutoshh

COPY hello.sh /code/

CMD bash /code/hello.sh
```

Container Restart Policy

`no`	Do not automatically restart the container. (the default)
`on-failure[:max-retries]`	Restart the container if it exits due to an error, which manifests as a non-zero exit code. Optionally, limit the number of times the Docker daemon attempts to restart the container using the `:max-retries` option.
`always`	Always restart the container if it stops. If it is manually stopped, it is restarted only when Docker daemon restarts or the container itself is manually restarted. (See the second bullet listed in restart policy details)
`unless-stopped`	Similar to `always`, except that when the container is stopped (manually or otherwise), it is not restarted even after Docker daemon restarts.

Container registry

- An image library place
- Store personal and professional images
- For sharing images to other docker users
- Listed Registries
  - Hosted :- Docker hub, quay.io etc
  - Cloud based: ECR , ACR , OCR , GCR
  - On prim : docker based , jfrog , harbor

Registry image format

Examples:

- Docker hub : docker.io/username/repo:tag
- Quay : quay.io/username/repo:tag

Cgroups

- To restrict resources for containers
- We can limit resources to containers like RAM,CPU,NETWORK,IO
- Setting cpu core
  - –cpuset-cpus {0-3}
  - –cpu-shares {0-1-24}
- Memory restrictions
  - RAM – –memory