

Jenkins

What is DockerFile:

- A Dockerfile is a text file that contains instructions for building a Docker image
- A Dockerfile typically contains a set of instructions that specify how to create a new image based on an existing base image.

create Jenkins in dockerfile:

```
FROM jenkins/jenkins:2.393
LABEL maintainer="mstewart@riotgames.com"

USER root
RUN mkdir /var/log/jenkins
RUN chown -R jenkins:jenkins /var/log/jenkins
USER jenkins
```

- The first line specifies the base image that the new image should be built from.
- The second line adds a label to the image that specifies the maintainer's email address.
- The next three lines switch to the root user and create a new directory for storing Jenkins logs (`RUN mkdir /var/log/jenkins`). They then change the ownership of the log directory to the Jenkins user. Jenkins has the necessary permissions to write to the log directory.
- The final line are executed with the correct user permissions.

Create docker-compose for jenkins:

```
version: '3.8'
services:
  jenkins:
    build:
      context: ../docker
      dockerfile: dockerfile
    privileged: true
    restart: always
    user: root
    ports:
      - 8080:8080
      - 50000:50000
    container_name: jenkins
    volumes:
      - ~/jenkins:/var/jenkins_home
      - /var/run/docker.sock:/var/run/docker.sock
      - /usr/local/bin/docker:/usr/local/bin/docker
```

- **version: '3.8':** This specifies the version of the Docker Compose file format to use.
- **services:** This is a section where you can define the different services that make up your application.
- **jenkins:** This is the name of the Jenkins service being defined.
- **build:** This specifies that the Docker image for the Jenkins service should be built from a Dockerfile in the "../docker" directory.
- **privileged: true:** This grants the container privileged access to the host system
- **restart: always:** This specifies that the container should always be restarted if it stops
- **user: root:** This sets the user that the Jenkins process will run as to the root user.

- `container_name : jenkins`: This sets the name of the container to "jenkins".
- three volumes from the host system to the container:
- `~/jenkins:/var/jenkins_home`: This maps the Jenkins home directory on the host to the container's Jenkins home directory. This allows Jenkins data to persist even if the container is destroyed.
- `/var/run/docker.sock:/var/run/docker.sock`: This maps the Docker socket on the host to the container's Docker socket. This allows the Jenkins container to interact with the host's Docker daemon.
- `/usr/local/bin/docker:/usr/local/bin/docker`: This maps the Docker binary on the host to the container's Docker binary. This allows the Jenkins container to execute Docker commands.