

Docker Image Deployment and Kubernetes Configuration

Your Name

November 1, 2024

Instructions

Prerequisites: Docker, Kubernetes, BusyBox, and OpenSSL installed, along with a working directory containing a Dockerfile.

1. Run a Local Docker Registry (without authentication):

```
docker run -d -p 5000:5000 --name registry
registry:2
```

This command starts a registry container on localhost:5000.

2. Optional: Enable Authentication for the Local Registry

If you want to secure your local registry with a username and password, follow these steps:

- (a) **Create a User and Password Hash with BusyBox:** Generate a hashed password for 'passpass' and store it in the 'htpasswd' file:

```
mkdir -p /home/user/auth
echo "user:${(busybox_mkpasswd -m sha256 -S $(
openssl_rand -hex 6)_passpass)}" > /home/user
/auth/htpasswd
```

This command generates a hashed password 'passpass' for the user-name 'user'.

- (b) **Run the Registry with Authentication:** Restart the registry container to require authentication.

```
docker stop registry
docker rm registry
docker run -d -p 5000:5000 --name registry \
-v /home/user/auth:/auth \
-e "REGISTRY_AUTH=htpasswd" \
```

```
-e "REGISTRY_AUTH_HTPASSWD_REALM=Registry_
Realm" \
-e "REGISTRY_AUTH_HTPASSWD_PATH=/auth/
htpasswd" \
registry:2
```

3. Build and Push Docker Image to Local Registry:

```
docker build -f Dockerfile -t localhost:5000/my-
app:latest ..
docker push localhost:5000/my-app:latest
```

4. Configure Kubernetes to Use the Local Registry (if using authentication): Create a Kubernetes secret with credentials to access the registry:

```
kubectl create secret docker-registry regcred \
--docker-server=localhost:5000 \
--docker-username=user \
--docker-password=passpass \
--docker-email=user@example.com
```

This secret can be referenced by Kubernetes to pull the image from the secured local registry.

5. Deploy a Pod Using the Image from the Local Registry:

```
kubectl run my-app --image=localhost:5000/my-app:
latest --image-pull-policy=IfNotPresent
```