

To use a container registry with MicroK8s, you can follow these steps:

To use a container registry with MicroK8s, you can follow these steps:

1. Enable the `registry` add-on in MicroK8s by running the following command:

```
```bash
microk8s enable registry
```
```

This command will start a local container registry in your MicroK8s cluster.

2. Verify that the `registry` add-on is running by checking the status:

```
```bash
microk8s status | grep registry
```
```

The output should show `enabled` next to the `registry` add-on.

3. Push an image to the local container registry. First, tag an existing Docker image with the address of the local registry. For example:

```
```bash
docker tag my-image:latest localhost:32000/my-image:latest
```
```

Replace `my-image` with the name of your image. The address `localhost:32000` corresponds to the default address of the local registry in MicroK8s.

4. Push the tagged image to the local container registry:

```
```bash
docker push localhost:32000/my-image:latest
```
```

This will push the image to the local registry within MicroK8s.

5. Use the image from the local registry in your Kubernetes manifests or deployments. Update your YAML files to reference the image from the local registry. For example:

```

```yaml
apiVersion: v1
kind: Pod
metadata:
 name: my-pod
spec:
 containers:
 - name: my-container
 image: localhost:32000/my-image:latest
```

```

Replace `my-image` with the name of your image, and `localhost:32000` with the address of the local registry.

MicroK8s will now use the local container registry for pulling images within your cluster.

Please note that the local container registry in MicroK8s uses port `32000` by default. If you encounter any issues or conflicts with this port, you can customize the port by modifying the `registry` add-on configuration.

By following these steps, you can enable and use a local container registry with MicroK8s to push and pull container images within your cluster.

To resolve this issue, you can configure Docker to allow insecure connections to the local registry by following these steps:

1. Open the Docker configuration file (`daemon.json`) for editing. The location of this file depends on your operating system:

- Linux: `/etc/docker/daemon.json`
- Windows: `C:\ProgramData\Docker\config\daemon.json`
- macOS: `~/Library/Group Containers/group.com.docker/daemon.json`

2. If the file doesn't exist, create it.

3. Add the following content to the `daemon.json` file:

`json` Copy code

```

{
  "insecure-registries": ["192.168.178.100:32000"]
}

```

Replace `192.168.178.100:32000` with the IP address and port of your local container registry.

4. Save the changes to the `daemon.json` file.
5. Restart the Docker daemon for the changes to take effect. The procedure for restarting the Docker daemon varies depending on your operating system. You can usually do this by restarting the Docker service or using the Docker desktop application.

6. `sudo snap restart docker.dockerd`

Once Docker is configured to allow insecure connections, you should be able to push images to the local container registry without encountering the HTTPS error.

Revision #3

Created 26 May 2023 05:18:44 by Alois

Updated 13 June 2023 12:45:13 by Alois