KUBERNETES SECRET FOR DOCKER REGISTRY

Cognitive Assistant for Networks (CAN) Release 6.0

JANUARY 23, 2023

AVANSEUS TECHNOLOGY PVT. LTD.

Revision History

Version	Date	Change description	Created by	Updated by	Reviewed by
V 1.0	Jan, 2023	Initial Release	Hemanth/Umesh	Raksha	Chiranjib

Table of Contents

1. Background

During the testing phase of any project we usually make several continuous updates to the base code as per the changing customer requirements and need to deploy the latest updated build in the kubernetes cluster. So this document targets to address below scenarios:

1. Pushing images to the docker registry
2. Configuring secrets in the helm charts

This document assumes that the Docker registry has already been created and now the interested images have to be pushed to the existing docker registry and the same image name has to be used in the helm charts.

We know that we are referring to the docker image in the 'values.yaml' file of our Helm charts. Hence for the successful installation of the helm chart we need to configure secrets correctly inside our kubernetes cluster and in values.yaml file of helm chart so that upon installation of the helm chart the intended docker image is pulled from the designated docker repository.

1.1. Pushing Images to the Docker registry

Prerequisites:

You must know the Docker registry details, such as:

- IP address or domain name where the registry has been created.
- Docker login credentials.

Note: If you haven't created the Docker registry, then please refer https://docs.docker.com/registry/ to install and configure the Docker registry. It is important to note that throughout this document we will refer to **avanseuscontainer.com** which will be used only by R & D team. The delivery team has to create their own Docket registry.

Follow the steps to push the images to the Docker registry:

1. Login to the Docker Registry.

   $docker login avanseuscontainer.com

   Enter the password of the Docker registry.

2. Create Docker Image

   Follow the Docker images creation documentation for further details. After all the images have been built, the images are ready to be pushed to the registry.

   Use the below command to see all the image:

   $sudo docker images

```
[ec2-user@ip-172-31-23-138 ~]$
[ec2-user@ip-172-31-23-138 ~]$ sudo docker images
REPOSITORY                                                  TAG       IMAGE ID       CREATED          SIZE
avanseuscontainer.com/vmware/5.0/predictioncontrollercppsimd   v1        f75fad6ca343   14 hours ago     448MB
predictioncontrollercppsimd                                 1         f75fad6ca343   14 hours ago     448MB
avanseuscontainer.com/vmware/5.0/predictioncontrollerboth   v1        305c9304d8f9   15 hours ago     672MB
predictioncontrollerboth                                    v1        305c9304d8f9   15 hours ago     672MB
avanseuscontainer.com/vmware/5.0/predictionworkerjava       v1        45d056f305a4   15 hours ago     660MB
predictionworkerjava                                        v1        45d056f305a4   15 hours ago     660MB
avanseuscontainer.com/vmware/5.0/predictioncontrollerboth   <none>    67d4f74054f2   3 days ago       672MB
avanseuscontainer.com/vmware/5.0/predictioncontrollerboth   <none>    0ca9d4355c00   4 days ago       672MB
avanseuscontainer.com/vmware/5.0/predictionworkerjava       <none>    126e3f552f98   4 days ago       660MB
```

3. Tag the built images.

   Use the below command to tag the built images:

```
$sudo docker tag <image_name>
avanseuscontainer.com/vmware/5.0/predictionworker:1
```

Example:

```
$sudo docker tag ldap:1 avanseuscontainer.com/vmware/5.0/predictionworker:1
```

4. Push the tagged images into the registry.

```
$sudo docker push avanseuscontainer.com/vmware/5.0/predictionworker:1
```

All the Docker images have been pushed to the registry and are ready to be used.

Follow the same steps to push all the Docker images to the Docker registry.

1.2. Configuring Secrets in the Helm Charts

Prerequisites:

- Docker Registry Credentials
- Fully qualified domain name of our private Docker registry.

For creating Kubernetes secrets, you should know the username and password of the docker registry.

Follow the steps to configure secrets in the Helm Chart:

1. To create Kubernetes secrets, you should know the username and password of the Docker registry. The username and password have to be encoded in the base64 encoding scheme before creating the secrets.

   Example: We have a Docker registry at **avanseuscontainer.com.**

   The username and password of the registry have to be encoded with a base64 encoding scheme.

   Take the username and password and write in the form "**username:password**" i,e., "**test:testpwd**".

2. Encode the above using the link provided.
3. Encoded output of "**test:testpwd**" is "**dGVzdDp0ZXN0cHdk**".
4. The encoded output will be used to create the kubernetes secrets.
5. To create the kubernetes secrets, create a file named "config.json"
6. Open the file using vi editor and add the following contents:

```
{
   "auths": {
      "<registry domain/ip>": {
         "auth": "<encoded credentials>"
      }
   }
}
```

Example:

```
{
   "auths": {
      "avanseuscontainer.com": {
         "auth": "dGVzdDp0ZXN0cHdk"
```

```
    }
  }
}
```

7. To create the secret, run the below command in the master node.

```
$ kubectl create secret generic <secret_name>
--from-file=.dockerconfigjson=config.json
--type=kubernetes.io/dockerconfigjson -n avanseus-workspace
```

<secret_name>: Any name, same name has to be specified in all the values.yaml file of all the helm charts in the pull secret part.

Example:

```
$ kubectl create secret generic pull_secret
--from-file=.dockerconfigjson=config.json
--type=kubernetes.io/dockerconfigjson -n avanseus-workspace
```