



Apache Airflow Deployment

To start the deployment, we need to be logged into the cluster:

```
# Replace the command with your own one inside the single quotes and run the cell
# Example OC_LOGIN_COMMAND='oc login --token=sha256~3bR5KXgwiUoaQiph2_kIXCDQnVfm_HQy3YwU2m-l
OC_LOGIN_COMMAND='oc login --token=sha256~OEGyKEhNUw7pHu_we-Js5YLz_9aSdImWlMGejETRuqc --serv
$OC_LOGIN_COMMAND
```

In order to identify and separate Airflow from the rest of the cluster, we create a project called `airflow`

```
oc new-project airflow
```

Now, we get the official helm charts for Airflow:

```
helm repo add apache-airflow https://airflow.apache.org
helm repo update
```

The next commands avoid security errors that cause deployment failures.

```
oc adm policy add-scc-to-group anyuid system:serviceaccounts:airflow
oc adm policy add-scc-to-group privileged system:serviceaccounts:airflow
```

And the actual deployment starts now:

```
helm upgrade --install airflow apache-airflow/airflow --namespace airflow
```

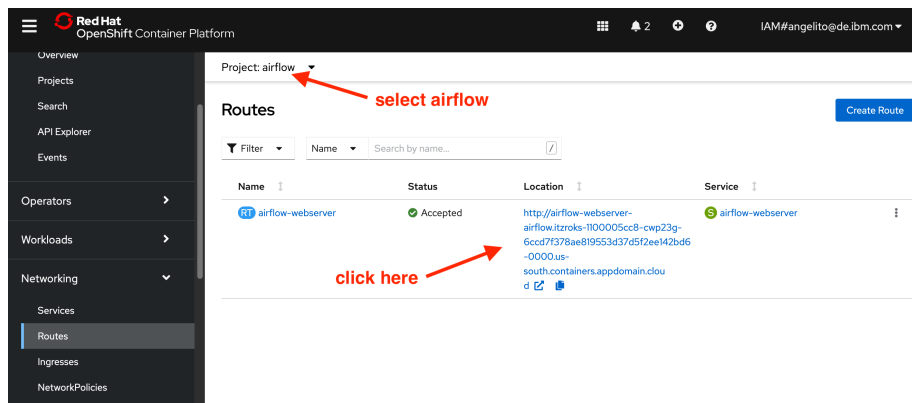
If the deployment went well, you will see several pods running:

```
oc get pods
```

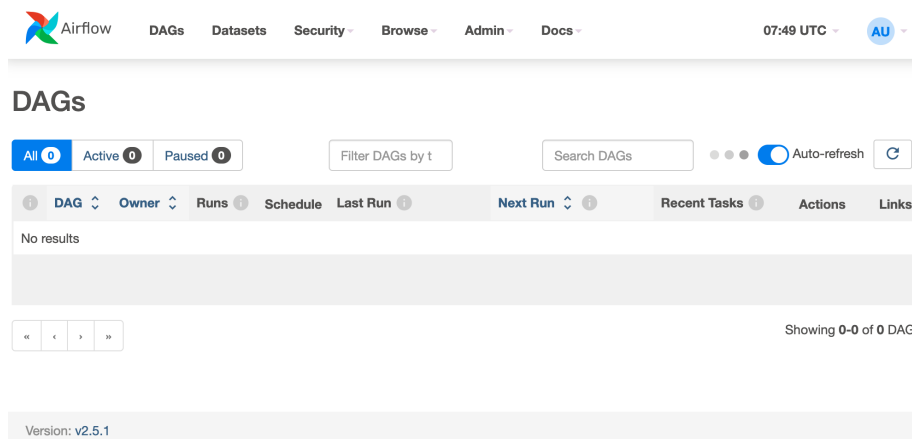
Next, you we need to add a route to the cluster to access Airflow:

```
oc get svc
oc expose svc airflow-webserver
```

And, just like we did with databand, we get access url from the OpenShift console:



You can login with the default user: **admin** and default password: **admin**. Then you will see the main dashboard of Airflow



Indeed, the main dashboard is empty because the default helm deployment disables the default DAGs. If you really miss them, you can modify an environment and re-deploy Airflow

```
helm upgrade airflow apache-airflow/airflow -f - << EOF
extraEnv: |
  - name: AIRFLOW__CORE__LOAD_EXAMPLES
    value: 'True'
EOF
```

And now, you will see them:

