Airflow Installation

- Postgres Setup: https://argoid.atlassian.net/wiki/spaces/ARGOID/pages/1335623681/PostgreSQL+Installtion
- Airflow libraries to be installed on Python3 virtual environment , so Python3 virtual-environment need to be created before airflow installation
  `sudo yum install -y epel-release`

  `sudo yum install -y python36  python36-devel python36-libs python-pip gcc gcc-c++`

  `sudo pip install  virtualenv`
  `sudo mkdir -p /opt/python3_6-virtual_envs`

  `sudo virtualenv -p python3 /opt/python3_6-virtual_envs/airflow_env` (or) sudo python3 -m venv /opt/python3_6-virtual_envs/airflow_env

- Create airflow user in a system
  `sudo useradd airflow`
  `sudo chown -R airflow:airflow /opt/python3_6-virtual_envs/airflow_env`
- switch to `airflow` user
  `sudo su - airflow`
- Airflow Installation
  `source /opt/python3_6-virtual_envs/airflow_env/bin/activate`

```
pip install apache-airflow['postgres']
```

- Postgres Tables creation
  Login postgresql shell

```
CREATE DATABASE airflow;
CREATE USER airflow WITH PASSWORD 'airflow';
GRANT ALL PRIVILEGES ON DATABASE airflow TO airflow;
```

- Provide PostgreSQL db information in `airflow.cfg` file , modify the `sql_alchemy_conn` property in `~/airflow/airflow.cfg` configuration file
  `sql_alchemy_conn = postgresql+psycopg2://airflow:airflow@<postgres_vm_ip>/airflow`

```
#   SequentialExecutor , LocalExecutor , CeleryExecutor , DaskExecutor ,
# ``KubernetesExecutor``, ``CeleryKubernetesExecutor`` or the
# full import path to the class when using a custom executor.
executor = SequentialExecutor

# The SqlAlchemy connection string to the metadata database.
# SqlAlchemy supports many different database engine, more information
# their website
#sql_alchemy_conn = sqlite:////data/1/airflow/airflow.db
sql_alchemy_conn =  postgresql+psycopg2://airflow:airflow@10.0.0.19/airflow
# The encoding for the databases
sql_engine_encoding = utf-8

# Collation for ``dag_id``, ``task_id``, ``key`` columns in case they have different encoding
# This is particularly useful in case of mysql with utf8mb4 encoding because
# primary keys for XCom table has too big size and ``sql_engine_collation_for_ids`` should
# be set to ``utf8mb3_general_ci``.
# sql_engine_collation_for_ids =
```

- Initialize the airflow database
  Remember: All the airflow commands to be executed in python3 virtual env
  `airflow db init`
- Start airflow-webserver on required port number
  `airflow webserver -p 8082` (or) airflow webserver -p 8082 > webserver_log.log 2>&1 &

- Create airflow web users
  `airflow users create --username admin --firstname Peter --lastname Parker --role Admin --email spiderman@superhero.org`

  Provide a new password, in the password prompt
  Now Login to Airflow UI with URL *http://<airflow_vm_ip>:8082*
- Start airflow scheduler
  `airflow scheduler` (or) airflow scheduler > scheduler_log.log 2>&1 &

Connecting Spark and Hive

- `sudo yum install hive.noarch`
- `sudo cp /usr/lib/hive/lib/hive-hcatalog-core-2.3.2.jar /opt/spark/jars/`
- Copy `hive-site.xml` file from the VM where Hive Services are running to the location `/opt/spark/conf/`

For spark operator

```
source /opt/python3_6-virtual_envs/airflow_env/bin/activate
pip3 install apache-airflow-providers-apache-spark
```

To Enable Airflow Monitoring: Monitoring Airflow using Prometheus