Install AirFlow

1.Install Apache Airflow on Ubuntu on AWS

Airflow is one of the most popular workflow management solution, it author, schedule and monitor workflows.

Requirements

- Python 2.7
- pip
- Ubuntu 16

2.Install Python, pip, Airflow and dependencies

First we need to install `python` and the python package management tool `pip`.

By default , for ubuntu python package are already been installed.

To install dependencies follow below steps.

```
###################################

sudo apt-get install python-setuptools

pip is a Python package management tool. We'll be using this for
installing packages required in Airflow.

To install pip:

sudo apt-get install python-pip

Note: It is recommended to use latest pip version. For upgrading pip
version, use pip upgrade command given below.

sudo pip install --upgrade pip
```

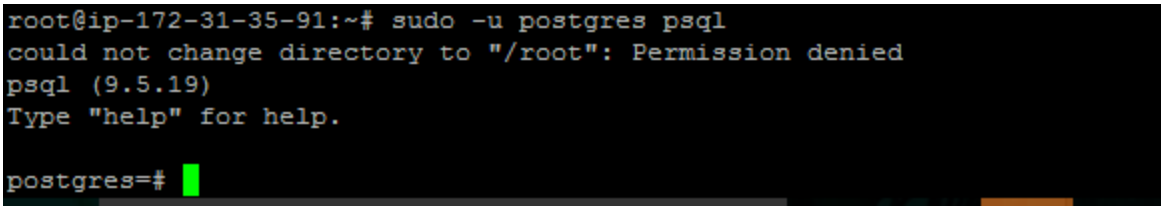
3. Installing PostgreSQL for Airflow

Airflow is shipped with a sqlite database backend. This database system will not be able to run data pipeline on webUI. We would require to have more powerful database system like PostgreSQL, it is an open source database management system, that comes with robust feature set, data integrity and extensibility. We will install PostgreSQL and configure it to use with Airflow.

```
sudo apt-get install postgresql postgresql-contrib
```

create a database for airflow and grant access to a sudo user. Lets access to psql, a command line tool for Postgres.

```
sudo -u postgres psql
```

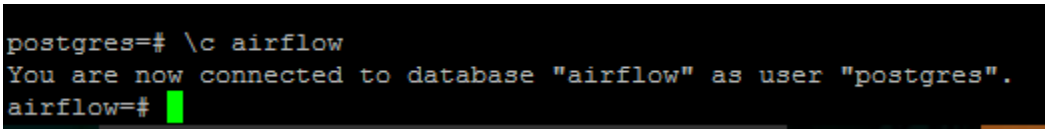


After logging in successfully, we will get psql prompt (postgres=#). We will create a new user and provide privileges to it.

```
CREATE ROLE ubuntu;
CREATE DATABASE airflow;
GRANT ALL PRIVILEGES on database airflow to ubuntu;
ALTER ROLE ubuntu SUPERUSER;
ALTER ROLE ubuntu CREATEDB;
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public to ubuntu;
```

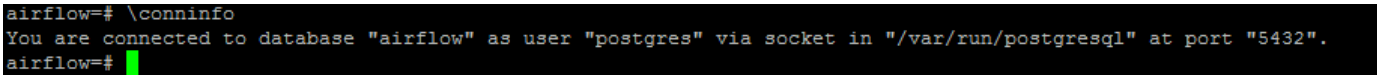
Now connect to airflow database and get connection information.

```
postgres-# \c airflow
```



After successful connection, prompt will be changed to airflow-#. We will verify this by fetching connection info

```
airflow=# \conninfo
```



We'll change settings in pg_hb.conf file for required configuration as per Airflow. You can run command SHOW hba_file;

```
vi /etc/postgresql/9.5/main/pg_hba.conf

change ipv4 address to 0.0.0.0/0

And restart PostgreSQL to load changes.

sudo service postgresql restart
```

4. Install Airflow

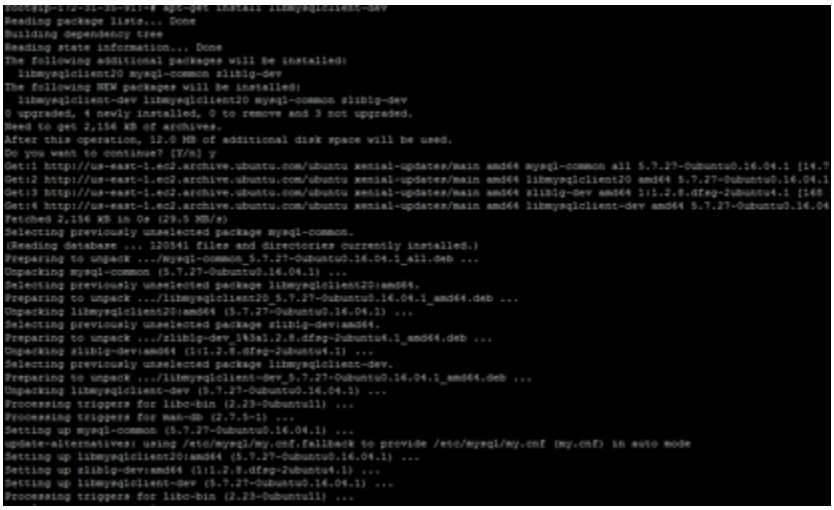
As PostgreSQL is already installed and configured. Next, We will install Airflow and configure it.

Set AIRFLOW_HOME environment variable to ~/airflow.

```
export AIRFLOW_HOME=~/airflow
```

First install the following dependencies:

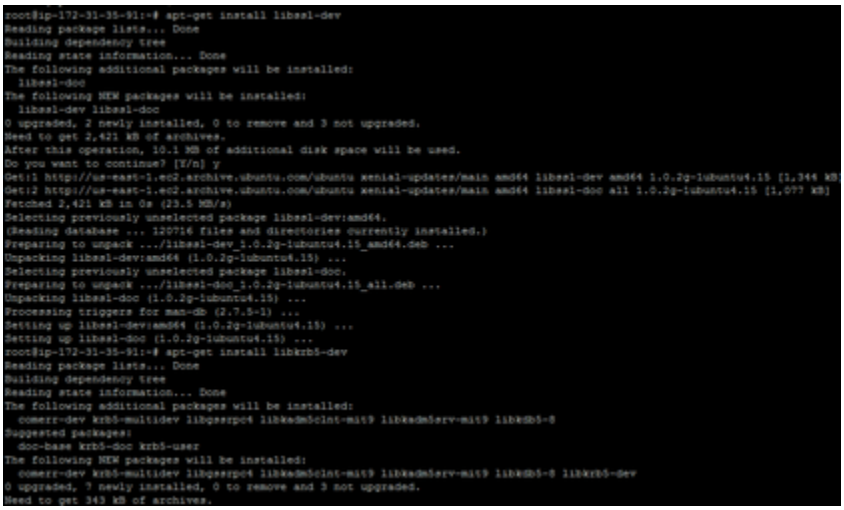
- `sudo apt-get install libmysqlclient-dev` (*dependency for airflow[mysql] package*)



- `sudo apt-get install libssl-dev` (*dependency for airflow[cryptograph] package)*

```
root@ip-172-31-35-91:~# apt-get install libssl-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libssl-doc
The following NEW packages will be installed:
  libssl-dev libssl-doc
0 upgraded, 2 newly installed, 0 to remove and 3 not upgraded.
Need to get 2,421 kB of archives.
After this operation, 10.1 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libssl-dev amd64 1.0.2g-1ubuntu4.15 [1,344 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libssl-doc all 1.0.2g-1ubuntu4.15 [1,077 kB]
Fetched 2,421 kB in 0s (23.5 MB/s)
Selecting previously unselected package libssl-dev:amd64.
(Reading database ... 120716 files and directories currently installed.)
Preparing to unpack .../libssl-dev_1.0.2g-1ubuntu4.15_amd64.deb ...
Unpacking libssl-dev:amd64 (1.0.2g-1ubuntu4.15) ...
Selecting previously unselected package libssl-doc.
Preparing to unpack .../libssl-doc_1.0.2g-1ubuntu4.15_all.deb ...
Unpacking libssl-doc (1.0.2g-1ubuntu4.15) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up libssl-dev:amd64 (1.0.2g-1ubuntu4.15) ...
Setting up libssl-doc (1.0.2g-1ubuntu4.15) ...
root@ip-172-31-35-91:~#
```

- *`sudo apt-get install libkrb5-dev` (dependency for airflow[kerbero] package)*

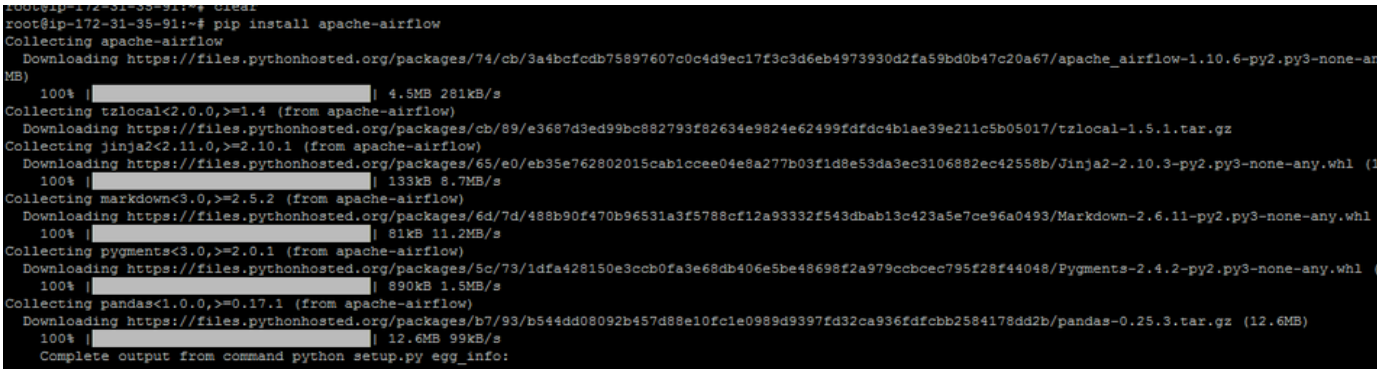


- *`sudo apt-get install libsasl2-dev` (dependency for airflow[hive] package):*

```
Processing triggers for libc-bin (2.23-0ubuntu11) ...
root@ip-172-31-35-91:~# apt-get install libsasl2-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  libsasl2-dev
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 254 kB of archives.
After this operation, 831 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libsasl2-dev amd64 2.1.26.dfsg1-14ubuntu0.1 [254 kB]
Fetched 254 kB in 0s (2,266 kB/s)
Selecting previously unselected package libsasl2-dev.
(Reading database ... 122659 files and directories currently installed.)
Preparing to unpack .../libsasl2-dev_2.1.26.dfsg1-14ubuntu0.1_amd64.deb ...
Unpacking libsasl2-dev (2.1.26.dfsg1-14ubuntu0.1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up libsasl2-dev (2.1.26.dfsg1-14ubuntu0.1) ...
root@ip-172-31-35-91:~#
```

After installing these dependencies, we can install airflow and its packages. (You can modify these packages depending on need. Celery and RabbitMQ are needed to use the Web-based GUI)

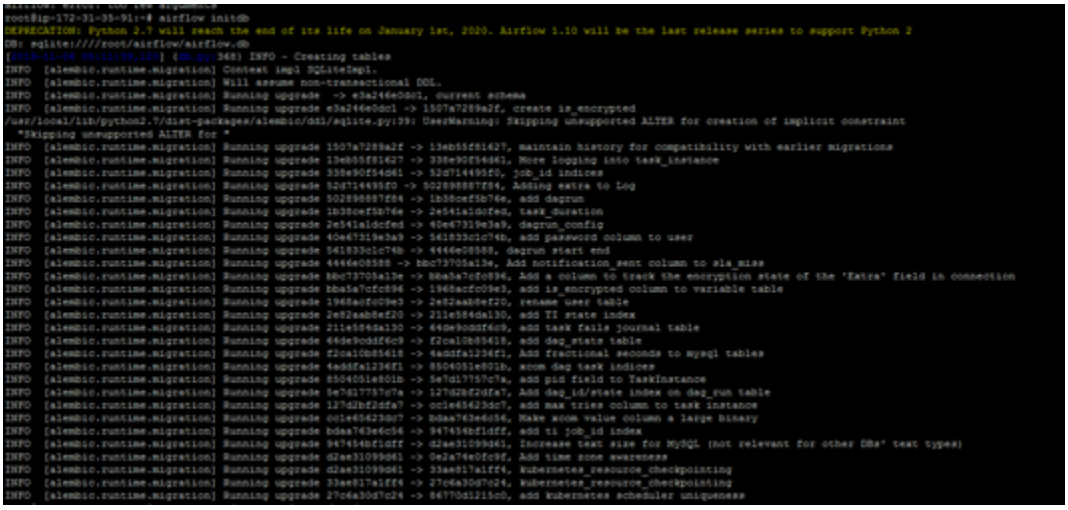
```
pip install apache-airflow
```



for other subpackages like celery, async, crypto, rabbitmq etc., you can check apache airflow installation page.

After successfully installing airflow, we will initialise Airflow's database

```
airflow initdb
```



To set up the first-time configs. An `airflow.cfg` file is generated in the airflow home directory.

```
##########################
home directory location

/root/airflow


##########################
```

```
root@ip-172-31-35-91:~# cd /root/airflow
root@ip-172-31-35-91:~/airflow# ll
total 124
drwxr-xr-x 3 root root  4096 Nov  6 05:11 ./
drwx------ 8 root root  4096 Nov  6 05:16 ../
-rw-r--r-- 1 root root 31200 Nov  6 05:11 airflow.cfg
-rw-r--r-- 1 root root 72704 Nov  6 05:11 airflow.db
drwxr-xr-x 3 root root  4096 Nov  6 05:11 logs/
-rw-r--r-- 1 root root  2501 Nov  6 05:11 unittests.cfg
root@ip-172-31-35-91:~/airflow#
```

We should open it with a text editor, and change some configurations in the [core] section:

- For the executor, we should use CeleryExecutor instead of SequentialExecutor if we want to run the pipeline in the webUI:

```
# The executor class that airflow should use. Choices include
# SequentialExecutor, LocalExecutor, CeleryExecutor, DaskExecutor, KubernetesExecutor
#executor = SequentialExecutor
executor = CeleryExecutor
```

- For the backend DB connection, we should pass along the connection info of the postgresql database `airflow` we just created:

```
##################################################

sql_alchemy_conn =
postgresql+psycopg2://airflow:airflow@ip-172-31-35-91:5432/airflow

load_examples = false
```

After doing all these setting just save your configuration and exit.

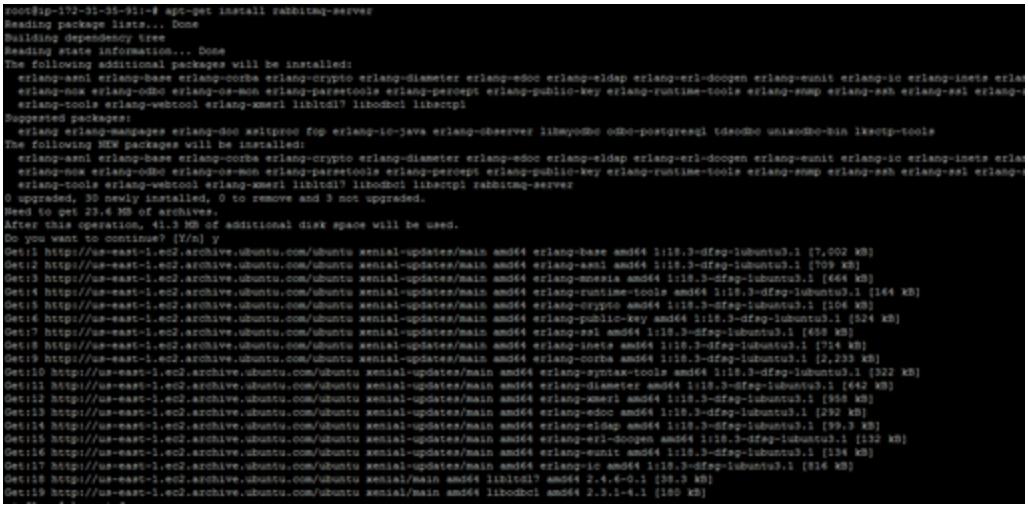
For Loading new configurations, we should run

airflow initdb

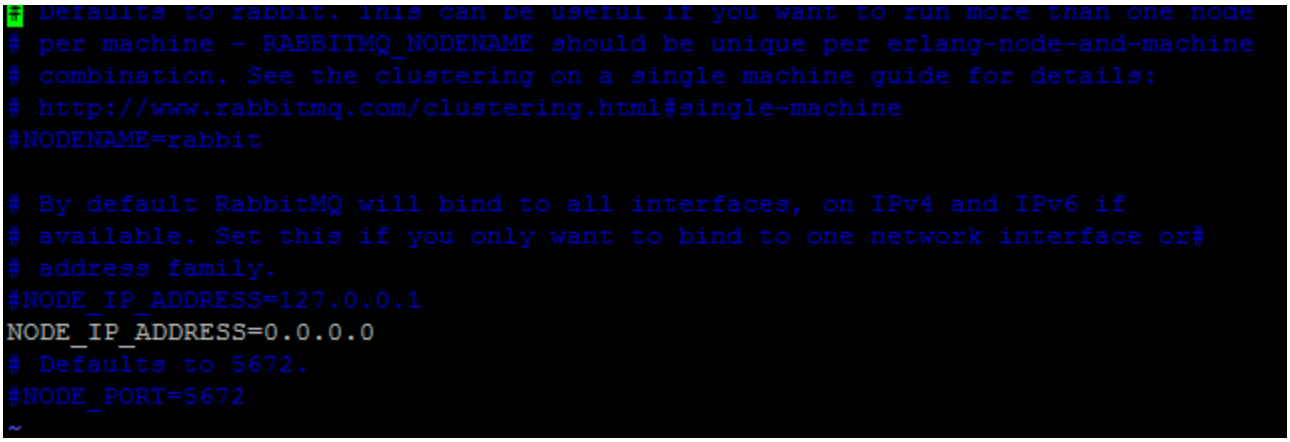
5.Installing Rabbitmq

Rabbitmq is a message broker, that required to rerun airflow dags with celery. Rabbitmq can be installed with following command.

```
sudo apt-get install rabbitmq-server
```



change configuration NODE_IP_ADDRESS=0.0.0.0 in configuration file located at **/etc/rabbitmq/rabbitmq-env.conf**



Then start rabbitmq service

```
sudo service rabbitmq-server start
```

6.Installing Celery

Celery is a python api for rabbitmq, We can install celery using pip

```
sudo pip install celery
```

7.Starting Airflow

All the required installation and configuration is done. We will create a dags folder in airflow home directory .i.e; at /home/ubuntu/airflow location

```
mkdir -p /home/ubuntu/airflow/dags/
```

and then we'll start all airflow services to up airflow webUI

```
airflow webserver
airflow scheduler
airflow worker
```

If you want to up airflow continuously up, you should run these command with -D flag like airflow webserver -D, this will run airflow as a Daemon in background. You required to do it for all the services, If you want to keep these services continuously up.

- Once webserver is up once with ip address with default port 8080
  http://ec2-3-87-103-89.compute-1.amazonaws.com:8080/admin/

  Dashboard for airflow: