**Wikipedia Data Pipeline using Airflow, EC2, S3**

* Creating a data pipeline to extract data from a Wikipedia page and store it in S3 bucket using python and orchestrating with airflow.
* Extracted the data using BeautifulSoup and created a dataframe using pandas. Turned this program into a function
* Stored the data as a csv file.
* Created an ec2 instance for running airflow.
* Created a python script (DAG) importing the etl code for extraction as a function and saving it.
* Checking the DAG in airflow.
* Instead of saving as csv can change the path to s3. But bucket must be created first.

**Details:**

For extraction, check the github repo in SuyaByte about etl extraction with python.

Save the data as a dataframe and then save it as a csv file. Create a folder for all the files and keep the files only in that folder.

Now we have to launch an ec2 instance. Click launch instance, give a name. Select Ubuntu. Select instance type (t2.micro for free tier)

**For key pair (login):**

Key pair name: select create new key pair, give a name, then select create.

It will download the key. Copy it in the folder created which has the etl python code and csv file.

In network, check all 3 options for ssh, http and https traffic.

Launch instance. Select on instance link, you’ll go to instances.

Then click on pending, status will change to running after some time.

Select the checkbox of the instance, then click connect.

Go to ssh client for the connection, copy the command which aws generates.

In terminal go to the folder of the code with key, then enter the command and connect.

**Run few commands to install and update few packages:**

sudo apt-get update

sudo apt install python3-pip

sudo pip install apache-airflow

sudo pip install pandas

#sudo pip install s3fs

#sudo pip install tweepy

sudo pip install bs4

Whatever are needed to run the python code.

enter airflow to check if it is installed.

enter “airflow standalone”.

once you see airflow is ready, copy the username and password. You’ll need it to login into airflow after launching it from ec2 instance.

{Login with username: admin

password: NV6y9PceFAh92ckF}

go to the instance in ec2, copy public ipv4 dns.

paste the url in browser: ec2-3-138-34-185.us-east-2.compute.amazonaws.com and add :8080

ec2-3-138-34-185.us-east-2.compute.amazonaws.com:8080

enter (if it is not running make sure you use http:// instead of https://) This part didn’t run. Could be because of t2.micro as it is only 1gb. Maybe airflow needs 4gb. Also checked for failure fixes in stackoverflow. This is the link: <https://stackoverflow.com/questions/10253484/cant-access-site-on-ec2-instance-via-public-ip>

**unable to connect to airflow server, why? ^**

Could be the security groups.

Go to the instance, scroll down, go to security, click on security groups, click edit inbound rules, then add rule, All traffic, IPV4 anywhere or myip, save. (not a good practice)

**Let’s assume the issue is solved and we are able to launch airflow.**

To create an S3 bucket, select create bucket, enter globally unique bucket name, enter create, check if it is created. To store data in s3 mention the path in first python script (etl script not dag) as “s3://bucket-name/filename.csv”

Now open new terminal as old terminal has server running, go to the airflow folder, make connection with the ec2 again for ssh client connection,

**Make some changes to airflow**

Do ls in airflow folder, there is a file airflow.cfg, edit it using sudo nano airflow.cfg,

Dags\_folder: instead of dags change it to twitter\_dag

Save it.

Create the twitter\_dag folder using mkdir, in the twitter\_dag folder paste the etl python file and dag python file

Check if the dags are showing up in airflow and resolve in case of any issues. (one issue could be naming mismatches across the file/folder names vs the paths in cfg file)

Stop the airflow server. CTRL+C.

un airflow as standalone again.

Go to graph, press run at the right side.

Error: when calling create bucket access denied. Means we don’t have permission to write from ec2 to s3 bucket. Need to set up those permissions. Go to ec2 instance. Go to actions at right side. Click security. Click modify IAM role [1].

Create new IAM role.

In roles, select create role. Select aws service (1st option). Select ec2 for common use cases. Click next. In Add permissions, search bar, click s3. Select AmazonS3FullAccess. Next type EC2 and select AmazonEC2FullAccess. Click next. Give a name for the role. Click create role.

Now go to the EC2>instances>selected instance>modify IAM roll at [1], In choose IAM role drop-down you can see the newly created role and select Update IAM role. This should give permission to ec2 to access s3 bucket.

Check the status of the dag in airflow.

To validate go to s3 and check if the bucket has the file with extracted data.