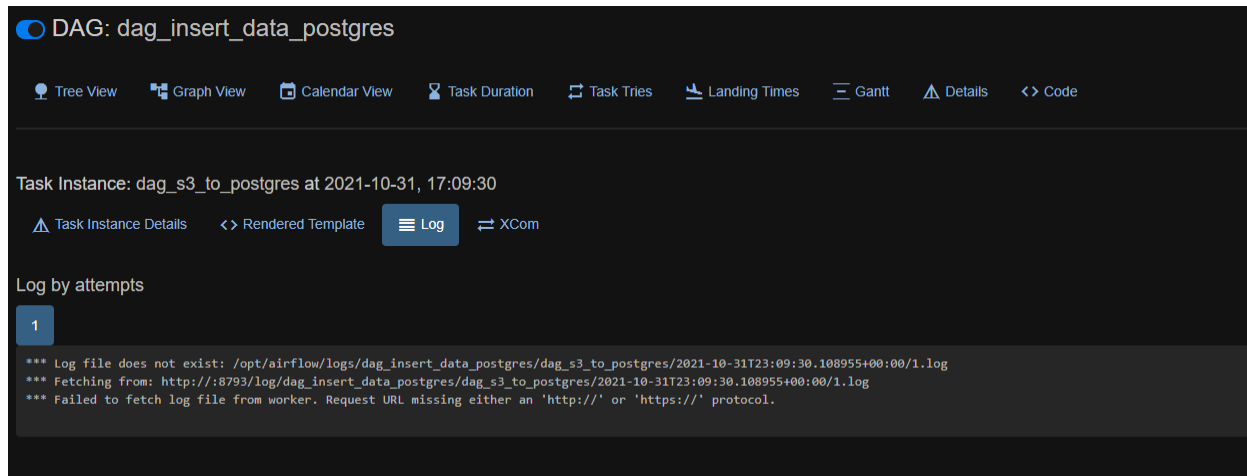## Data Engineering Bootcamp 2nd Deliverable

Although I wrote the code for this deliverable, I was not able to run the main task of creating a table in Postgres with the given data. The main problem was with the airflow logs. I don't know if the airflow version used had a bug but I was getting the message on the following image (trying to get the file from an S3 bucket).

This error message prevented me from debugging the DAG efficiently. I did search on google how to fix it but I could not find a solution. To help me fix the problem I made the DAG smaller to something which I was able to run correctly. Adding some code sometimes did not work and would end up with the message on the first image. After some time, I was able to get some log messages which gave me what I needed to fix some bugs on my code.

The following image has the error which up to this moment I have been unable to fix. It seems to be a problem with the connection to the S3 bucket.
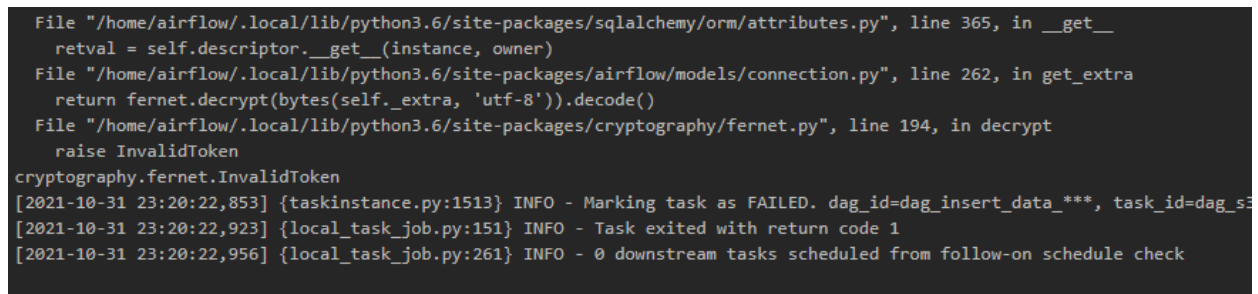
Checking out Airflow's documentation I found out that the fernet key was an environment variable. I tried to modify through the helm chart without success. I thought of using Python's OS module to assign the correct value to the variable. The result was that the DAG failed to run giving the message from the first image in this document.

Code used to set the fernet key value without success.

```python
from cryptography.fernet import Fernet
import os


fernet_key = Fernet.generate_key()
os.environ["AIRFLOW__CORE__FERNET_KEY"] = fernet_key.decode()
```

Inside Airflow's UI I sometimes get the following message.

Ooops!

```
Something bad has happened.
Please consider letting us know by creating a bug report using GitHub.

Python version: 3.6.15
Airflow version: 2.1.4
Node: airflow-webserver-5fb75659f-xljbg
-------------------------------------------------------------------
Traceback (most recent call last):
  File "/home/airflow/.local/lib/python3.6/site-packages/flask/app.py", line 2447, in wsgi_app
    response = self.full_dispatch_request()
  File "/home/airflow/.local/lib/python3.6/site-packages/flask/app.py", line 1952, in full_dispatch_request
    rv = self.handle_user_exception(e)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask/app.py", line 1821, in handle_user_exception
    reraise(exc_type, exc_value, tb)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask/_compat.py", line 39, in reraise
    raise value
  File "/home/airflow/.local/lib/python3.6/site-packages/flask/app.py", line 1950, in full_dispatch_request
    rv = self.dispatch_request()
  File "/home/airflow/.local/lib/python3.6/site-packages/flask/app.py", line 1936, in dispatch_request
    return self.view_functions[rule.endpoint](**req.view_args)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask_appbuilder/security/decorators.py", line 109, in wraps
    return f(self, *args, **kwargs)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask_appbuilder/views.py", line 608, in edit
    widgets = self._edit(pk)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask_appbuilder/baseviews.py", line 1233, in _edit
    form = self.edit_form.refresh(obj=item)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask_appbuilder/forms.py", line 329, in refresh
    form = self(obj=obj)
  File "/home/airflow/.local/lib/python3.6/site-packages/wtforms/form.py", line 208, in __call__
    return type.__call__(cls, *args, **kwargs)
  File "/home/airflow/.local/lib/python3.6/site-packages/flask_wtf/form.py", line 87, in __init__
    super(FlaskForm, self).__init__(formdata=formdata, **kwargs)
  File "/home/airflow/.local/lib/python3.6/site-packages/wtforms/form.py", line 274, in __init__
    self.process(formdata, obj, data=data, **kwargs)
  File "/home/airflow/.local/lib/python3.6/site-packages/wtforms/form.py", line 126, in process
    if obj is not None and hasattr(obj, name):
  File "/home/airflow/.local/lib/python3.6/site-packages/sqlalchemy/orm/attributes.py", line 365, in __get__
    retval = self.descriptor.__get__(instance, owner)
  File "/home/airflow/.local/lib/python3.6/site-packages/airflow/models/connection.py", line 262, in get_extra
    return fernet.decrypt(bytes(self._extra, 'utf-8')).decode()
  File "/home/airflow/.local/lib/python3.6/site-packages/cryptography/fernet.py", line 194, in decrypt
    raise InvalidToken
cryptography.fernet.InvalidToken
```

Some questions I have are:

How to set the fernet key?

How to use an older airflow version on the cloud?