

## INSTALACIÓN DE AIRFLOW EN LA RASPBERRY PI 3

Antes de comenzar la instalación debe actualizar su sistema operativo:

```
rodrigo@raspberrypi:~$ sudo apt update && sudo apt upgrade -y
Obj:1 http://security.debian.org/debian-security bullseye-security InRelease
Obj:2 http://deb.debian.org/debian bullseye InRelease
Obj:3 http://deb.debian.org/debian bullseye-updates InRelease
Obj:4 http://archive.raspberrypi.org/debian bullseye InRelease
Obj:5 https://download.docker.com/linux/debian bullseye InRelease
Leyendo lista de paquetes... 31%
```

Posteriormente instale python3 junto con pip:

```
sudo apt install python3 python3-pip -y
```

```
rodrigo@raspberrypi:~$ sudo apt install python3 python3-pip -y
Leyendo lista de paquetes... Hecho
Creando Árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
python3 ya está en su versión más reciente (3.9.2-3).
python3-pip ya está en su versión más reciente (20.3.4-4+rpt1+deb11u1).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
rodrigo@raspberrypi:~$
```

Instale las dependencias:

```
sudo apt install python3-dev libatlas-base-dev -y
```

```
rodrigo@raspberrypi:~$ sudo apt install python3-dev libatlas-base-dev -y
Leyendo lista de paquetes... Hecho
Creando Árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
libatlas-base-dev ya está en su versión más reciente (3.10.3-10).
python3-dev ya está en su versión más reciente (3.9.2-3).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
rodrigo@raspberrypi:~$
```

Navegue hasta la carpeta del proyecto:

```
rodrigo@raspberrypi:~$ cd ComputacionTF/Air_Flow/
rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$
```

Y cree un ambiente virtual de python:

```
sudo python3 -m venv AirFlow
```

```
rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$ sudo python3 -m venv AirFlow
rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$
```

Active el ambiente virtual:

```
source AirFlow/bin/activate
```

```
rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$ source AirFlow/bin/activate
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$
```

Configure la variable de entorno de la aplicación:

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

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$ export AIRFLOW_HOME=~/.airflow
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow$
```

Instale el programa AirFlow:

```
sudo pip3 install "apache-airflow==2.7.1" --constraint
```

```
"https://raw.githubusercontent.com/apache/airflow/constraints-2.7.1/constraints-3.9.txt"
```

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $ sudo pip3 install "apache-airflow==2.7.1" --constraint "https://raw.githubusercontent.com/apache/airflow/constraints-2.7.1/constraints-3.9.txt"
Traceback (most recent call last):
  File "/usr/bin/pip3", line 33, in <module>
    sys.exit(load_entry_point('pip==20.3.4', 'console_scripts', 'pip3')())
  File "/usr/bin/pip3", line 25, in importlib_load_entry_point
    return next(matches).load()
  File "/usr/bin/pip3", line 25, in <module>
    sys.exit(load_entry_point('pip==20.3.4', 'console_scripts', 'pip3')())
```

Inicialice la base de datos:

```
airflow db init
```

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $ airflow db init
/usr/local/lib/python3.9/dist-packages/airflow/cli/commands/db_command.py:43 DeprecationWarning: `db init` is deprecated. Use `db migrate` instead to migrate the db and/or airflow connections create default connections to create the default connections
DB: sqlite:///home/rodrigo/airflow/airflow.db
```

Cree un usuario con permisos de administrador:

```
airflow users create --role Admin --username
```

```
me rodrigo --password megamanzero --email samborms@gmail.com --firstname Admin
```

```
--lastname User
```

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $ airflow users create --role Admin --username me rodrigo --password megamanzero --email samborms@gmail.com --firstname Admin --lastname User
/usr/local/lib/python3.9/dist-packages/flask_limiter/extension.py:336 UserWarning: Using the in-memory storage for tracking rate limits as no storage was explicitly specified. This is not recommended for production use. See: https://flask-limiter.readthedocs.io#configuring-a-storage-backend for documentation about configuring the storage backend.
```

Cree el archivo dags:

```
mkdir ~/airflow/dags
```

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $ mkdir ~/airflow/dags
```

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $ ls
AirFlow
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $
```

Inicie el servidor de AirFlow:

```
airflow webserver -p 8080 &
```

Espere hasta que el servidor se active y se coloque en modo “espera y escuchando”:

```
(AirFlow) rodrigo@raspberrypi:~/ComputacionTF/Air_Flow $ sudo
Running the Gunicorn Server with:
Workers: 4 sync
Host: 0.0.0.0:8080
Timeout: 120
Logfiles: - -
Access Logformat:
=====
/usr/local/lib/python3.9/dist-packages/flask_limiter/extension.py:336 UserWarning: Using the in-memory storage for tracking rate limits as no storage was explicitly specified. This is not recommended for production use. See: https://flask-limiter.readthedocs.io#configuring-a-storage-backend for documentation about configuring the storage backend.
[2025-10-03 23:15:01 -0600] [1927] [INFO] Starting gunicorn 21.2.0
[2025-10-03 23:15:01 -0600] [1927] [INFO] Listening at: http://0.0.0.0:8080 (1927)
[2025-10-03 23:15:01 -0600] [1927] [INFO] Using worker: sync
[2025-10-03 23:15:01 -0600] [1930] [INFO] Booting worker with pid: 1930
[2025-10-03 23:15:02 -0600] [1931] [INFO] Booting worker with pid: 1931
[2025-10-03 23:15:02 -0600] [1932] [INFO] Booting worker with pid: 1932
[2025-10-03 23:15:02 -0600] [1933] [INFO] Booting worker with pid: 1933
```

En otra sesión de Bash ejecute nuevamente un ambiente virtual de python y de ahí inicie el administrador de Dags:

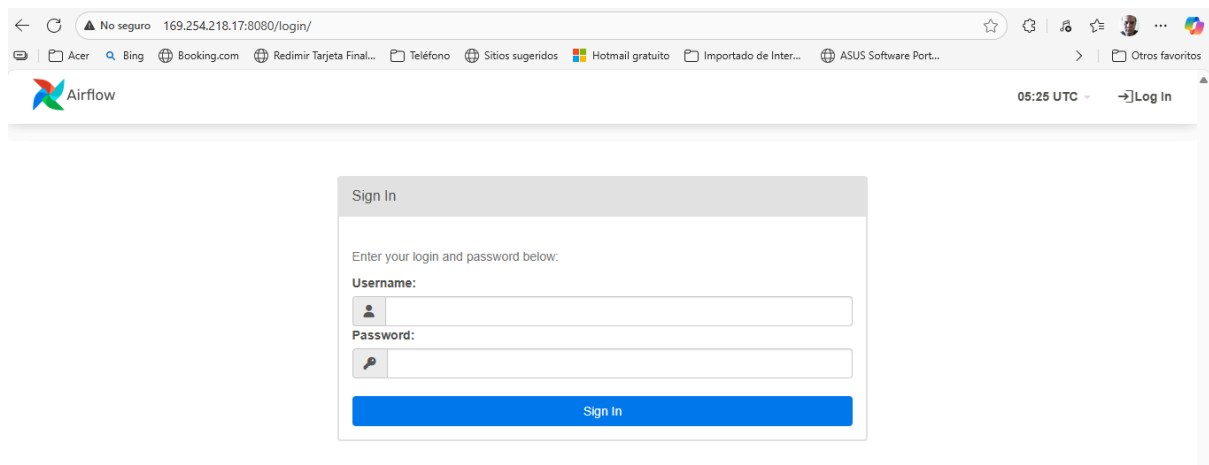
*airflow scheduler &*

```
rodrigo@raspberrypi: ~/ComputacionTF $ cd Air_Flow/
rodrigo@raspberrypi: ~/ComputacionTF/Air_Flow $ source .AirFlow/bin/activate
(.AirFlow) rodrigo@raspberrypi: ~/ComputacionTF/Air_Flow $ airflow scheduler &
[1] 1972
(.AirFlow) rodrigo@raspberrypi: ~/ComputacionTF/Air_Flow $

[2025-10-04T00:21:07.674-0500] [executor_loader.py:117] INFO - Loaded executor: SequentialExecutor
[2025-10-03 23:21:07 -0600] [1986] [INFO] Starting gunicorn 21.2.0
[2025-10-03 23:21:07 -0600] [1986] [INFO] Listening at: http://[::]:8793 (1986)
[2025-10-03 23:21:07 -0600] [1986] [INFO] Using worker: sync
[2025-10-03 23:21:07 -0600] [1988] [INFO] Booting worker with pid: 1988
[2025-10-04T00:21:08.064-0500] [scheduler_job_runner.py:798] INFO - Starting the scheduler
[2025-10-04T00:21:08.068-0500] [scheduler_job_runner.py:805] INFO - Processing each file at most -1
times
[2025-10-03 23:21:08 -0600] [1989] [INFO] Booting worker with pid: 1989
[2025-10-04T00:21:08.094-0500] [manager.py:166] INFO - Launched DagFileProcessorManager with pid: 1989
```

Abra el explorador web y conéctese al servidor local o desde una máquina conectada a la misma red:

<http://169.254.218.17:8080/>



Utilice el usuario y contraseña creados anteriormente para ingresar:

