



Creating a Hook to Interact With Elasticsearch

Interacting with an external system



The Plugin

During this section, we are going to code a plugin which will be composed of a Hook, an Operator as well as a Web View. The purpose of the plugin is to be able to connect with Elasticsearch (a search engine), make some interactions with it, and visualise its statistics directly from the Airflow UI.

This section is not meant to create a fully functional and perfect plugin for Elasticsearch with Apache Airflow. The goal is to show you how to code a plugin through an interesting example in order to give you the ability of coding your own plugin.

Enough talking and let's start coding!



Instructions

- `cd ~`
- `mkdir -p ~/airflow/plugins`
- `cp -r ~/airflow_files/plugins/elasticsearch_plugin/ ~/airflow/plugins/`
- `vim ~/airflow/plugins/elasticsearch_plugin/__init__.py`

Instructions

- From the Airflow UI, click on “Admin”, “Connections” and “Create”.
- We are going to create the connection to Elasticsearch which will be used by our Hook.
- Fill the fields with the same values and click on save.

Connection [create]

List Create

Conn Id	elasticsearch_default
Conn Type	
Host	localhost
Schema	
Login	
Password	
Port	9200
Extra	

Save Save and Add Another Save and Continue Editing Cancel

Instructions

- `vim ~/airflow/plugins/elasticsearch_plugin/hooks/elasticsearch_hook.py`
- `cp ~/airflow_files/plugin_hook_dag.py ~/airflow/dags/`
- `vim ~/airflow/plugin_hook_dag.py`
- On the Airflow UI, trigger the DAG “plugin_hook_dag” as follow:

First, click here



- Then, go to the “Graph View” and wait for the DAG to finish.

Instructions

Once the DAG has finished executing, if you click on the task “hook_es” and “View Log” you should see the following result:

```
[2018-11-15 02:45:12,206] {models.py:1569} INFO - Executing <Task(PythonOperator): hook_es> on 2018-11-15T02:45:06.107952+00:00
[2018-11-15 02:45:12,207] {base_task_runner.py:124} INFO - Running: ['bash', '-c', 'airflow run plugin_hook_dag hook_es 2018-11-15T02:45:06.107952+00:00 --job_id 3 --raw -s']
[2018-11-15 02:45:15,273] {base_task_runner.py:107} INFO - Job 3: Subtask hook_es [2018-11-15 02:45:15,266] {settings.py:174} INFO - setting.configure_orm(): Using pool set
[2018-11-15 02:45:16,791] {base_task_runner.py:107} INFO - Job 3: Subtask hook_es [2018-11-15 02:45:16,783] {__init__.py:51} INFO - Using executor LocalExecutor
[2018-11-15 02:45:17,584] {base_task_runner.py:107} INFO - Job 3: Subtask hook_es [2018-11-15 02:45:17,576] {models.py:258} INFO - Filling up the DagBag from /home/airflow/
[2018-11-15 02:45:17,787] {base_task_runner.py:107} INFO - Job 3: Subtask hook_es [2018-11-15 02:45:17,785] {cli.py:492} INFO - Running <TaskInstance: plugin_hook_dag.hook_
[2018-11-15 02:45:18,034] {logging_mixin.py:95} INFO - [2018-11-15 02:45:18,033] {base_hook.py:83} INFO - Using connection to: localhost

[2018-11-15 02:45:18,044] {logging_mixin.py:95} INFO - [2018-11-15 02:45:18,044] {base.py:83} INFO - GET http://localhost:9200/ [status:200 request:0.009s]

[2018-11-15 02:45:18,044] {logging_mixin.py:95} INFO - {'name': 'UpgZp3W', 'cluster_name': 'elasticsearch', 'cluster_uuid': '3Buc1FxrRser8WiAI9EkYg', 'version': {'number':
```



This is the result fetch from Elasticsearch
using our Hook