Orchestrating hybrid workflows with Apache Airflow

Ricardo Sueiras
Principal Developer Advocate, AWS
Regulation & Compliance

Heritage systems

Reduce complexity

Cost effective
<table>
<thead>
<tr>
<th>SqlToS3Operator</th>
<th>PythonOperator</th>
<th>AthenaOperator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Need to integrate networks between Cloud and remote (via VPN for example)</td>
<td>Need to integrate networks between Cloud and remote (via VPN for example)</td>
<td>Need to integrate networks between Cloud and remote (via VPN for example)</td>
</tr>
<tr>
<td>• Additional work required to enable connectivity (inbound/controlled networks)</td>
<td>• More complex DAGs</td>
<td>• Processing of data in Cloud (Athena)</td>
</tr>
<tr>
<td>• Processing of data in Airflow Worker</td>
<td>• Processing of data in Airflow Worker</td>
<td>• Create Athena Federated Queries (running as Lambda functions)</td>
</tr>
<tr>
<td></td>
<td>• Potential Apache Airflow anti-pattern</td>
<td>• Additional work required to enable connectivity (inbound/controlled networks)</td>
</tr>
</tbody>
</table>
Bridging this gap.

Using existing operators may require you to deploy a VPN solution to access remote locations.
Develop ETL application
Package as container
Push to Container Repository
Orchestrate running container
**KubernetesOperator**

- Need to integrate networks between Cloud and remote (via VPN for example)
- Need to build your ETL container image
- Good re-usability
- Need to provision and manage K8s clusters locally

**ECSOperator**

- Need to install local ECS Anywhere agent and create Amazon ECS clusters
- Not open source
- Need to build your ETL container image
- Good re-usability
- Processing of data locally
Run container on ECS Cluster

- **Outbound Network Connectivity**
- Outbound connectivity
- Local installation requirements
  - Docker engine
  - System Manager Agent
  - ECS Agent
- Supported Linux distributions
  - Ubuntu 18.04, 20.04
  - RHEL/CentOS 7/8
  - Fedora, Debian, openSUSE
- **NEW** Windows

Run containers in the Cloud
- Amazon EC2
- AWS Fargate
- AWS Wavelength
- AWS Localzone

Run containers locally
- AWS Outposts
- ECS Anywhere
Demo
(live, pray for me!)
Hybrid data pipeline

AWS Cloud

Data Lake

Remote Office

Compliance restrictions

Local MySQL database

Data Centre

Regulatory controls

Local MySQL database

Public Cloud provider

Migration

Local MySQL database

Amazon RDS MySQL
Data Engineer

- Create ETL scripts to be containerised
- Author and create DAGs (workflows) using standard ECS Operators
- Manage/Schedule/Monitor workflows
- View Workflow logs

Apache Airflow

Sysadmin/Ops (Cloud)

- Provision infrastructure (ECS Clusters, ECS Anywhere, Networking, Databases, Apache Airflow, etc)
- Container lifecycle management - package/deploy containers into container image repository
- Manage secrets, IAM policies, etc

Sysadmin/Ops (Local)

- Install/configure local ECS Anywhere
- Manage local resources on which ECS Anywhere is installed

Develop ETL Scripts

- AWS CodeCommit
- AWS CodeBuild

Develop DAGs

- AWS CodeBuild

Code Pipeline

- Push
- Build
- Deploy
- Test

AWS Cloud

- Amazon Elastic Container Service (ECS)
- Amazon ECR
- Amazon Managed Workflows for Apache Airflow worker nodes
- Amazon Managed Workflows for Apache Airflow
- AWS Secrets Manager
- Amazon RDS MySQL
- Amazon S3
- Amazon CloudWatch

ECS Cluster

- Task definition
- ETL Container image

ECS Resource

ETL Container image

Task definition

AWS CodePipeline

- CodeCommit
- CodeBuild

Local Environment

- MySQL database

ECS Anywhere

- Local MySQL database
Permissions

**Task Definition Role** (ecsTaskExecutionRole)
- IAM policies needed for your application

**Task Execution Role** (ecsInstanceRole)
- IAM policies needed for the hosts to run your Containers

**ECS Anywhere Task Execution Role**
- Like the Task Execution Role but for the ECS Anywhere agent

---

**Amazon ECS Cluster**
- Task definition
- ETL Container image

**ECS Cluster Resource**

---

**Amazon Managed Workflows for Apache Airflow**

---

**AWS Cloud**
- IAM policies needed to access AWS services from your DAGs
- IAM policies needed for your application
- IAM policies needed for the hosts to run your Containers
- Like the Task Execution Role but for the ECS Anywhere agent

---

**ECS Anywhere**

---

**AWS Secrets Manager**
- MWAA Execution Role
- IAM policies needed to access AWS services from your DAGs
- IAM policies needed for your application
https://github.com/094459/blogpost-airflow-hybrid
Thank you!

Ricardo Sueiras

https://www.linkedin.com/in/ricardosueiras

@094459