Software Used:
  – Packer (imaging)
  – Ansible (provisioning image or infrastructure)
  – awscli / cloudformation (mgmt / provisioning infrastructure)

Base Image:

us-east-1	xenial	16.04 LTS	amd64	ebs-ssd	2017112 1.1	ami-6a 2da510	aki-919 dcaf8
us-east-1	xenial	16.04 LTS	amd64	instanc e-store	2017112 1.1	ami-2c 33bb56	aki-919 dcaf8
us-east-1	xenial	16.04 LTS	amd64	hvm:eb s-ssd	2017112 1.1	ami-aa2ea6 d0	hvm ** (using)

** - version used to utilize micro instance for testing

Steps Taken:
  – Read & understand documentation for airflow installation/configuration (https://github.com/apache/incubator-airflow)
  – Configured AWS profile
  – Created 'aaron-test.pem' for system access (manual step)
  – Created a basic directory structure for application
      – airflow.json (packer configuration file; variables, builders, provisioners)
      – airflow.yml (ansible playbook for building image, includes core service airflow[all] installation &configuration with applicable package dependencies, etc.)
      – airflow_cfn.json (CF template configuration for AWS resources)
      – airflow_params.json (parameters for CF template)
 - Coding for cloudformation/ansible
      - intermediate builds/deploys for testing along the way

AWS Services used:
  – ASG
  – ELB
  – Ended up doing redis/mysql on the single host to save some time. Intended to

allow the service to scale up or out if needed by using
- ElasticCache
- RDS
- Configuration items/files
    - DB, Service(s)

Results:
- Core Infrastructure provisions without error
- ELB has 0 instances active
    - Airflow service isn't started, initdb & web server startup & QA ; seem to be the final steps however.