Write-up Project 1

I did a terraform project which is

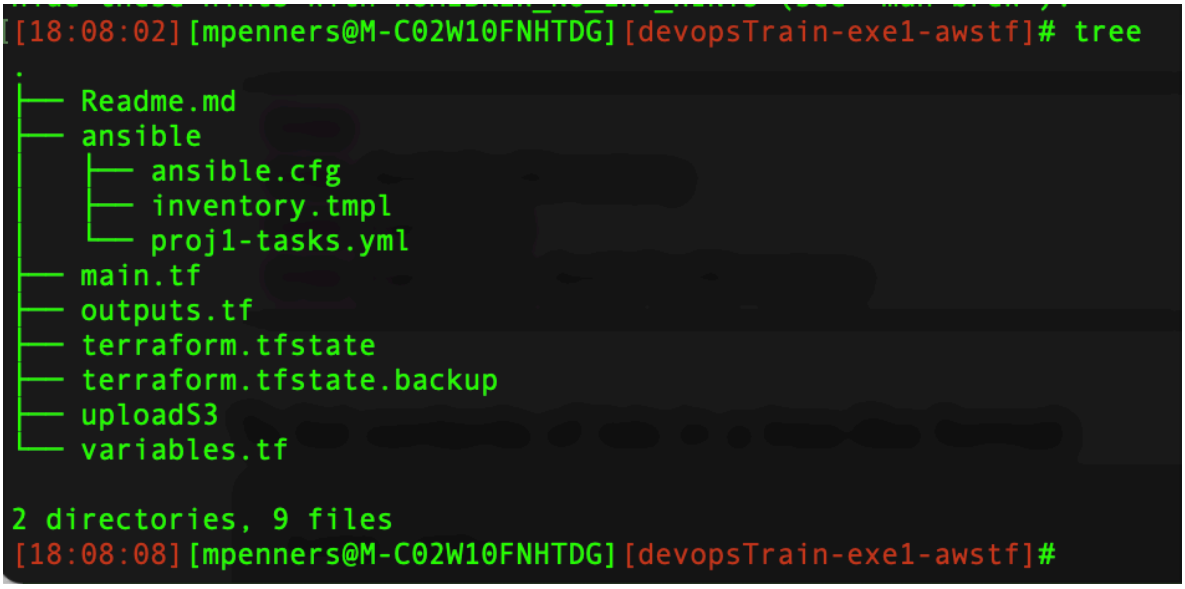
1. creating the EC2 instance and all it needs to connect to the internet
2. calling ansible to remote-install the required SW packages on the EC2 and start Jenkins.

Key thoughts are like this:

- All deployment specific information can be centrally set in variables.tf
- only terraform states to maintain the entire flow including the install
  - AWS provider to build the EC2 resources
  - required SW packages are installed via ansible to EC2
  - terraform should manage ansible calls from its variables.tf setting
  - terraform should manage ansible files to track status

The link to the repo is here:
git@github.com:pennersm/proj1--AWS-TF.git

Files in the repo:



ANSIBLE DIRECTORY:
ansible.cfg : can be used while testing a playbook from this directory
inventory.tmpl **:** will be rendered by terraform to include IP of the EC2 instance (for ansible hosts)
proj1-tasks **:** is the ansible playbook that installs the required SW packages

MAIN DIRECTORY:
main.tf **:** we start and end the complete project including the SW installations and starting Jenkins via this single main module
outputs.tf **:** not all variables are in here but some
.gitignore **:** it is not in this output but we use a gitignore with a template for

terraform. To prevent uploading large files or worse
variables.tf **:** central file to make all settings

Execution Screenshots and logs
You can follow the terraform steps to apply as described and documented in the readme file. Let me guide you through what happens when looking at the main.tf file:

1. terraform EC2 creation using aws provider is very straightforward, this goes up to line ~125
2. after ~125 lines it starts using other providers for the ansible hook. Comments along the resources:
    1. using resource local_file : this is to create the inventory file for ansible. its quick&dirty and does not scale for multiple groups. I will work on this using the provioner described in the main.tf
    2. using resource shell_script: this will invoke ansble during create phase and delete its leftover files including log durin the tf-destroy phase
3. you can find a ansible/current_ansible.log file during the EC2 is up that shows what has been installed by ansible. It will dissappear with tf destroy

1. the ansible_tf_state.json file supposedly creates an output that should be reflected in the terraform.tfstate file, showing the exit status of the ansible-playbook command. However, this is somehow not working (yet)