DESCRIPTION

Use Terraform to provision infrastructure

**Description:**

Nowadays, infrastructure automation is critical. We tend to put the most emphasis on software development processes, but infrastructure deployment strategy is just as important. Infrastructure automation not only aids disaster recovery, but it also facilitates testing and development.

Your organization is adopting the DevOps methodology and in order to automate provisioning of infrastructure there's a need to setup a centralised server for Jenkins.

Terraform is a tool that allows you to provision various infrastructure components. Ansible is a platform for managing configurations and deploying applications. It means you'll use Terraform to build a virtual machine, for example, and then use Ansible to instal the necessary applications on that machine.

Considering the Organizational requirement you are asked to automate the infrastructure using Terraform first and install other required automation tools in it.

**Tools required:** Terraform, AWS account with security credentials, Keypair

\*\*

Pre-Req:

1. Spin Windows 2019

-Launch a Windows 2019 Server of t2.micro instance type.

-Connect by downloading the rdp and accessing with decrypted password.

cjdVkS7wxyqCtCQuZ@q%U-B\*)=PT@;LC

-From Start> Sever Manager> Local Server >disable ‘IE Enhanced Security Configuration’

1. Download code from github <https://github.com/amikshas/DCT_Project1.git>
2. Download Terraform and place the executable in project folder:

-Download 386 from <https://www.terraform.io/downloads>

1. Install Visual studio code:

-Download visual studio code for Windows from <https://code.visualstudio.com/download>

-Install Extension for Terraform

1. Aws cli for windows <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>



**Architecture:-**

The terraform scripts spin 2 ubuntu machines with installing Ansible Controller in web-0

Once the instance is launched the Ansible playbook will install Jenkins, Java and Python in web-1

Webwin0 Webwin1

Ansible Controller

 Install Jenkins, Java, and Python

\*\*\*

Step1-Modify the ‘variable.tf’ script and update with the AWS access key, secret and session token

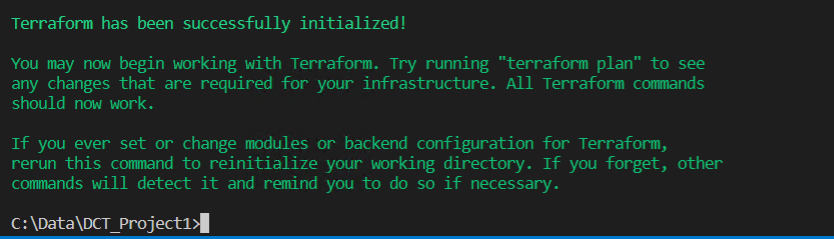
ASIA6D4UPZ55WE2MBF7X

lia+YW/1RmZ5BnZoCfVMABuuZKABWq+6E8UqhFDQ

FwoGZXIvYXdzEJb//////////wEaDOjvOczqzVqF95ly/CK6Ae5PjK6FTPXka0HOGUnk9KKOgj6lUikwPcwI8AZaFe89KmLmcf010LB0bWBjRqabbu5GhYDQuSOOm3E0P02dW5yLHCLZjuEzR39nE6+OeILOxUFaqS4BiUEfYy3kGtULG06cOcIp7Gfd9u+FP8YgwptClsLZ2ss1btVFZ52pwLbGTrU8PX091F7ZKcp+/NiVdDH8+JMAvQ/IKFGafDhFBmvHCXW8R6KiHi27Gg/R0MdNQvDrndBu6GWkAiio6KuYBjItNMjAa8yObGc1zJuYgo7QbwaKESkQxef7/DT7o9mXXm0sSshesqwckp7oUai0

Step2- Open the Project folder in Visual Studio Code and initialize terraform

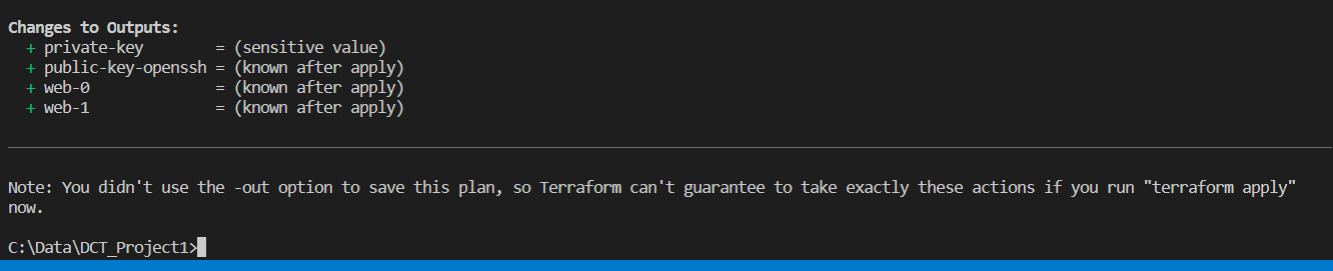
**terraform init**

Output:- 

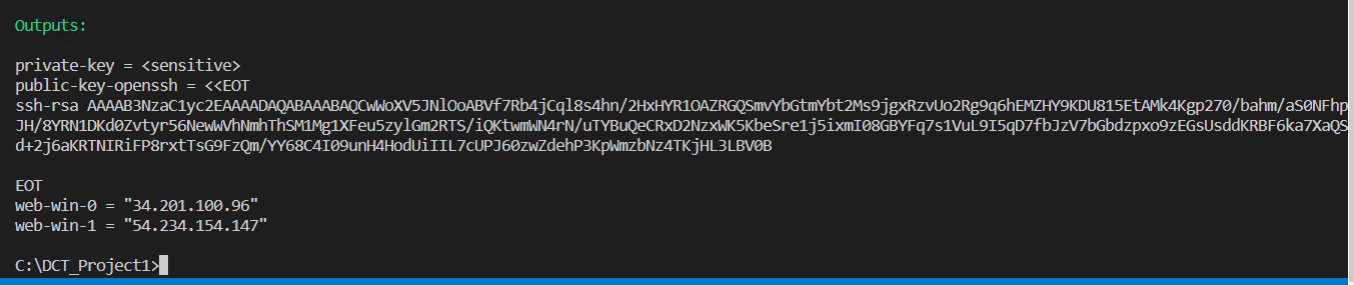
Step3- Plan terraform

**terraform plan**

Output:-

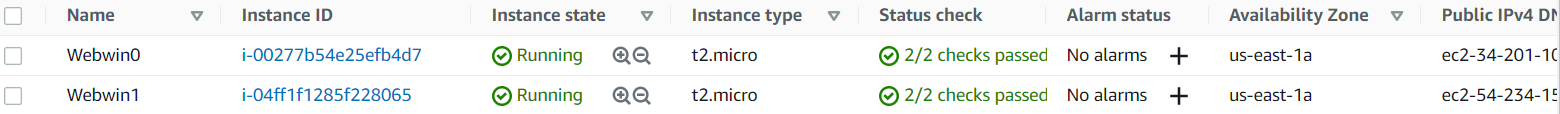


Step4:- apply terraform



Output:

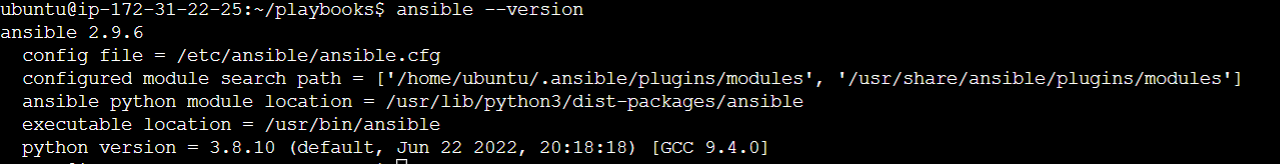
As above 2 ubuntu machines are spin as below:



\*\*

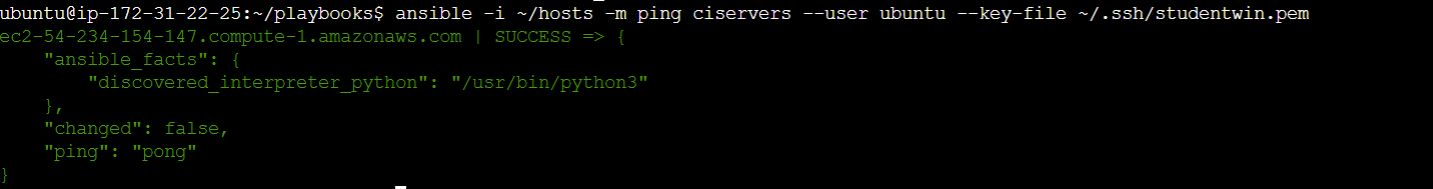
Step 5- Verify ansible is installed on Webwin0

**ansible --version**

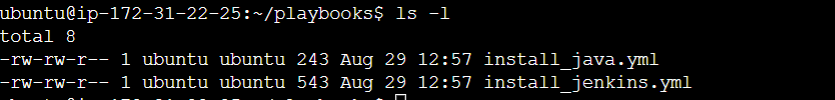


Step 6-Check the connectivity to remote machine Webwin1

**ansible -i ~/hosts -m ping ciservers --user ubuntu --key-file ~/.ssh/studentwin.pem**

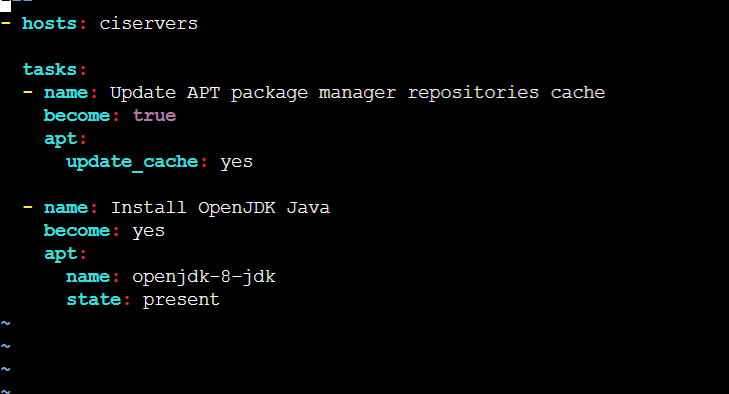


Step 7- Copy the playbook files as below in a folder

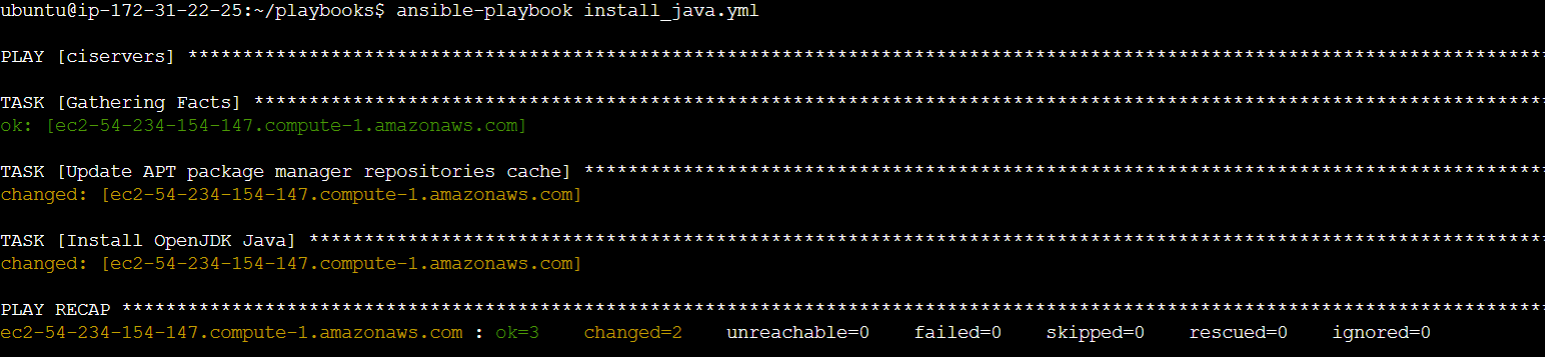


Step 8 – Install Java by running playbook install\_java.yml

**ansible-playbook install\_java.yml**



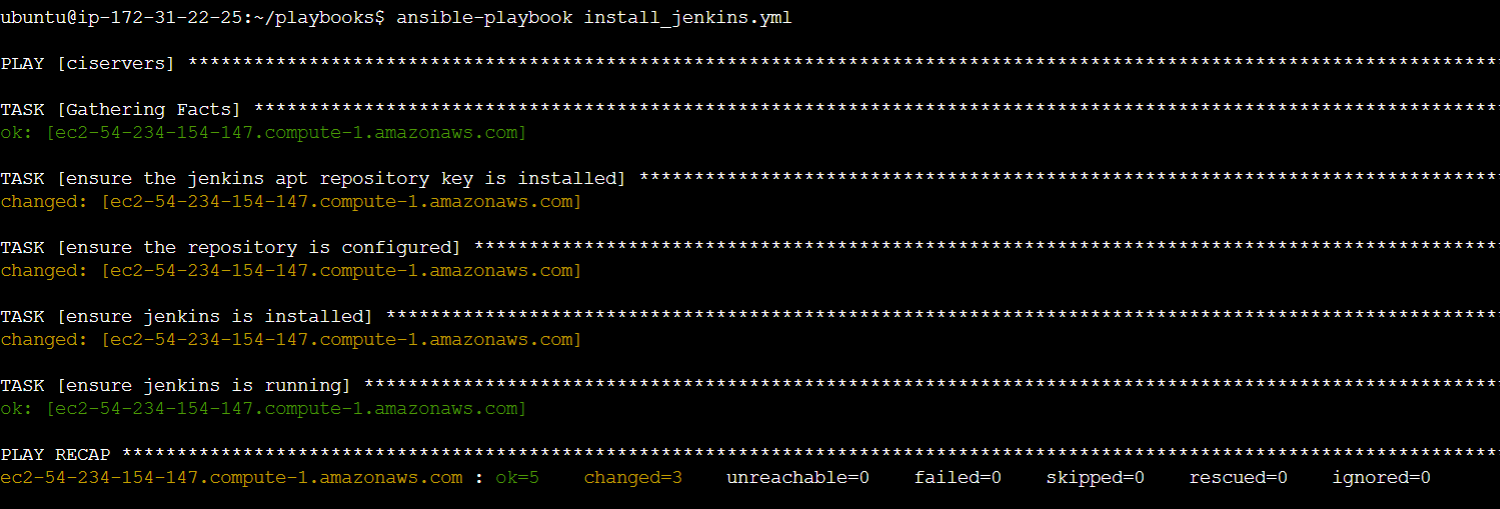
Output:-



Step 9 – Install Java by running playbook install\_jenkins.yml

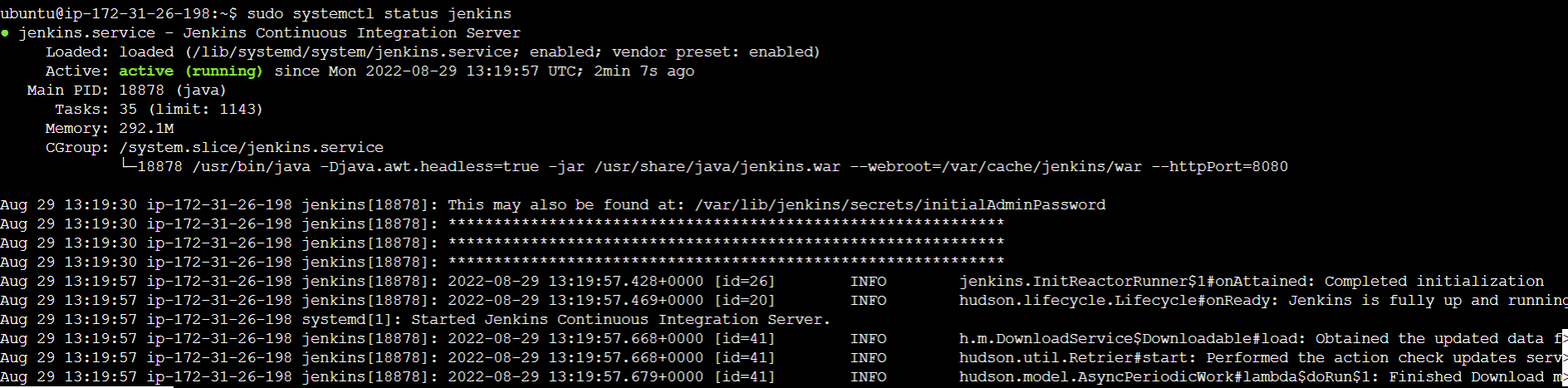
**ansible-playbook install\_jenkins.yml**

Output:



**Step 10:- Verify Jenkins is installed**

sudo systemctl status Jenkins



Load the url in browser and setup the initial like initial password and plugins setup of Jenkins

