**Agenda: Infrastructure automation using Terraform and Kubernetes cluster configuration using Ansible with Dynamic Inventory**

**Prerequisite:**

* AWS Account
* Create **Redhat** EC2 Instance
* Create IAM Roles with following permissions
  + VPCFullAccess
  + EC2FullAccess
  + S3FullAccess
* Attach IAM Role to EC2 Instance
* Create a Key Pair with the name as “<give-name>”

**Create User to Install Ansible and Terraform**

sudo useradd ansible

echo “ansible ALL=(ALL) NOPASSWD:ALL” | sudo tee /etc/sudoers.d/ansible

sudo su ansible

cd ~

**Install Terraform**

sudo yum install wget unzip -y

sudo wget [get the zip link from hashcorp site ]

sudo unzip [name of zip file] –d /usr/local/bin/ [please check the path on the system you are going to run this command for AWS instances it Is normally /bin/]

# add path permanently for current user by exporting path in .bashrc file at end of file.

vi .bashrc

export PATH=$PATH:/usr/local/bin/

# source .bashrc to reflect for current session

Source ~.bashrc

**Install Ansible**

sudo su ansible

sudo yum install python3 -y

sudo alternatives --set python /usr/bin/python3

sudo yum install python3-pip -y

pip3 install ansible --user

pip3 install boto3 --user # this will be used to fetch dynamic inventory using python, Python program will be written to fetch the AWS infrastructure

**Install git**

sudo yum install git -y

**Clone Terraform and Ansible scripts**

git clone https://github.com/RahulMishraAjmer/Terraform\_Kubernetes\_Cluster\_Ansible\_Dynamic\_Inventory.git

cd Terraform\_Kubernetes\_Cluster\_Ansible\_Dynamic\_Inventory

**Infrastructure as a Code**

# Initialise to install plugins

$ terraform init AWS\_VPC/

# Validate terraform scripts

$ terraform validate AWS\_VPC/

# Plan terraform scripts which will list resources which is going be created.

$ terraform plan AWS\_VPC/

# Apply to create resources

$ terraform apply --auto-approve AWS\_VPC/

**Set the region for AWS so that inventory.py may from which region the infrastructure details are to be fetched. You set the Environment variable on the machine from where you ae going to run this python file.**

export AWS\_DEFAULT\_REGION=ap-south-1

**Configuration Management**

Check if the dynamic inventory script works

$ chmod +x DynamicInventory.py

$ ./ DynamicInventory.py --list

Kubernetes Cluster setup using Ansible with Dynamic Inventory

$ vi devops.pem

$ chmod 400 devop.pem

Replace <pemfile> with your pemfile path in server

$ ansible-playbook –i DynamicInventory.py site.yaml –u ubuntu –private-key=<PemfilePath> --ssh-common-args “o StrictHostKeyChecking=no”

**Destroy Infrastructure**

$ terraform destroy --auto-approve VPC/