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 to automate provisioning of infrastructure there's a need to set up a centralized 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 install 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

**Expected Deliverables:**

Launch an EC2 instance using Terraform

Connect to the instance

Install Jenkins, Java, and Python in the instance

**Solution:**

Launch labs,

Devops in AWS – Linux environment to execute terraform code

AWS – Cloud environment to provision EC2 instances

Ensure the Linux environment that you are running is UpToDate

Ensure that your system is up to date, and you have the gnupg, software-properties-common, and curl packages installed.

sudo apt-get update && sudo apt-get install -y gnupg software-properties-common curl

sudo apt-get update

sudo apt-get install

Add the HashiCorp GPG key.

curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add –

Add the official HashiCorp Linux repository.

sudo apt-add-repository "deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb\_release -cs) main"

Update to add the repository and install the Terraform CLI.

sudo apt-get update && sudo apt-get install terraform

Verify the installation

Text

Description automatically generated

Install AWS CLI

Text

Description automatically generated

Set up AWS account and associated credentials that allow you to create resources.

Create IAM user to use with terraform

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generated

Configure environment variables

aws configure

AWS Access Key ID [\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WNBF]:

AWS Secret Access Key [None]: Rd1RWGoB7lbHh0P2nTbiAWl6aTuxo\*\*\*\*\*\*

Default region name [us-east-1]:

Default output format [None]:

Note the values of existing components like VPC and subnet to provision EC2

Vpc id - vpc-0445c7003cbc00c73

Subnet ID - subnet-096b04687ecd71826

Create key pair –

Graphical user interface, text, application

Description automatically generated

Create terraform file

Doc reference - <https://learn.hashicorp.com/tutorials/terraform/aws-change?in=terraform/aws-get-started>

Terraform init

Graphical user interface, text

Description automatically generated

Terraform apply

Graphical user interface, text

Description automatically generated