**Use case – 1: Build Landing zone (VPC, Subnets, Internet Gateway, Required IAM roles & Policies)**

**What is the learning from this use case :**

Now a days, in all the companies we heard the term called Landing zone, most of the people who had worked in Legacy Datacenter based projects know about this terminology; however let me explain in my own way.

Landing Zone – This is the base platform (In civil engineering, we call structure, basement with required no of pillars for your building) for your whole infrastructure, Applications.

This base platform includes,

* network setup – in public cloud we use the term VPC in AWS, vNET in Azure.

CIDR range, subnets, placement of subnets in each AZ, NAT Gateway, IG, IAM roles & Policies

* RBAC model – role based access control, you can create the resource access model based on each individual roles & responsibilities (Defined in IAM policies)
* Instance profile – This is used to access your EC2 instance securely within AWS browser, we no need to use the public IP to get the access to your EC2 instances.

In this video, I have explained how we can deploy these above-mentioned services in an AWS account using Terraform, to make it simple I have provided the very basic LZ components, however we would be able to add the required services/components based on the needs to design & deploy the LZ.

**Pre-requisites:**

1. Install Terraform executable on your local desktop/laptop
2. Install VS Code as a code editor for Terraform on your local desktop/laptop
3. Valid AWS account (AWS provides a 1 year valid free credits – refer AWS Free Tier – Types of offer – 12-months free)
4. Basic understanding of Terraform & AWS IaaS services

**Steps:**

* Login to AWS account with your account (it should have enough access to create the resources)
* Collect the Access key & Secret keys for the user (Secret key will not be visible after the first use, we should download and keep it safe, if we missed the secret key, we should generate the key once again and the same needs to updated in Terraform provider configuration)

**Simple Pictorial Representation:**

