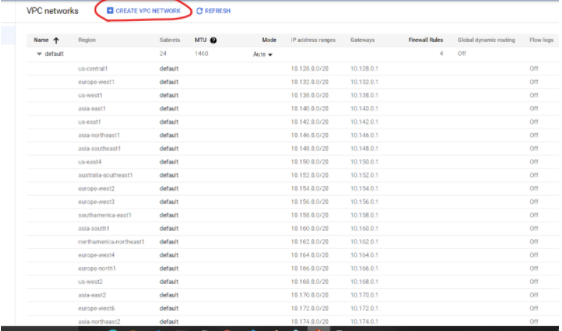
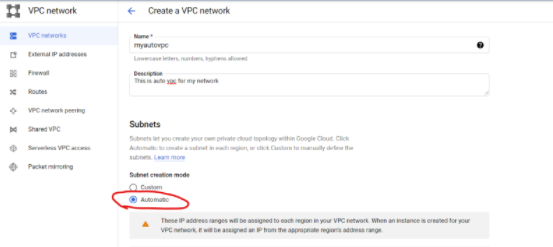
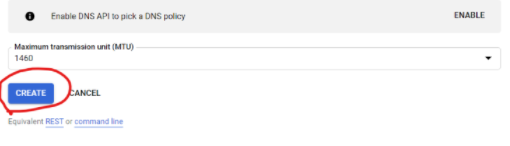
Google VPC (Virtual Private Cloud)

* VPC Provides networking functionality to Virtual Machines, GKE clusters and App Engine
* VPC can be thought in the same way as Physical Network, except it is virtualized within Google cloud
* VPC is a global resource i.e. it consists of list of regional subnetworks in datacenters, all connected by a Global wan.
* In GCP
  + vpc is a global resource
  + subnet is a regional resource
* While vpc is a gloabl resource, all routes and associated firewall rules are also global.
* All resources created within a vpc can talk to each other using the internal private IP address if you enable private google access to them.
* Subnets are associated with a particular region and in one region we can more than one subnet.
* VPC’s can be created in two modes
  + auto mode:
    - one subnet is automatically created inside each region using predefined IP ranges that fit with in a 10.128.0.0/9 block
  + custom mode: In this mode VPC network, no subnets are created and its up to you create your own subnets and IP ranges that best suits your needs.
* Every project in GCP will have one default VPC
* When creating an vm instance, you select a zone, a network and subnet
* There are some quotas(can be increased)/limits(cannot be increased) which can be increased by raising a ticket with Google.
  + VPC Networks per project Quota of 5
  + VM instances per network Limit of 7000

Lets Create an auto network in the project

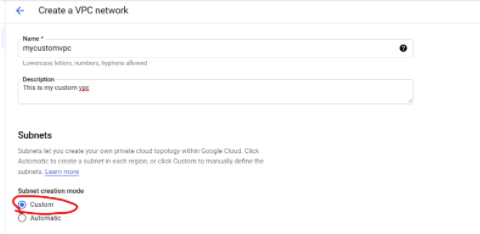


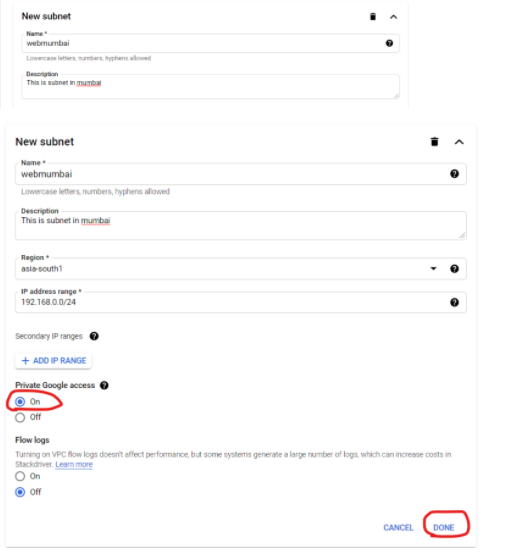


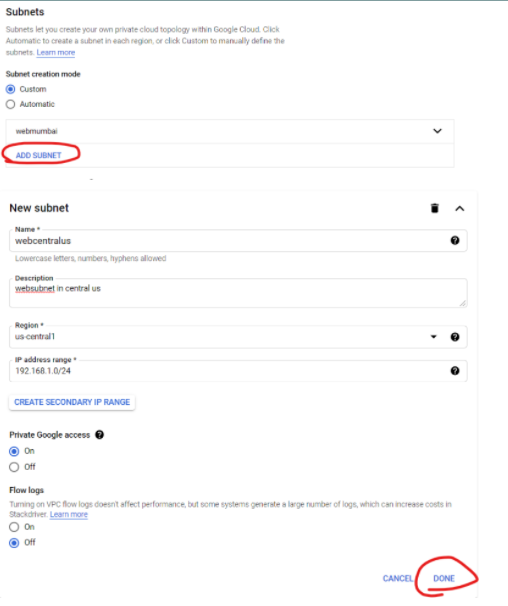


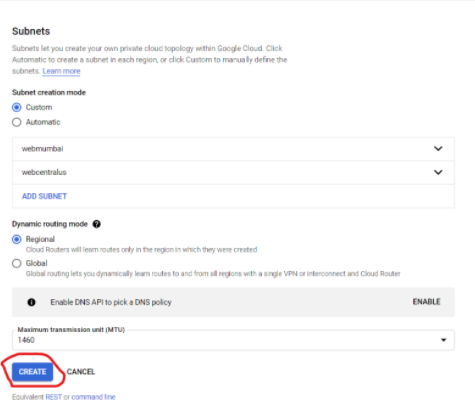
Lets create a vpc with range 192.168.0.0/16 and the subnets with 250 devices

* mumbai
* central us

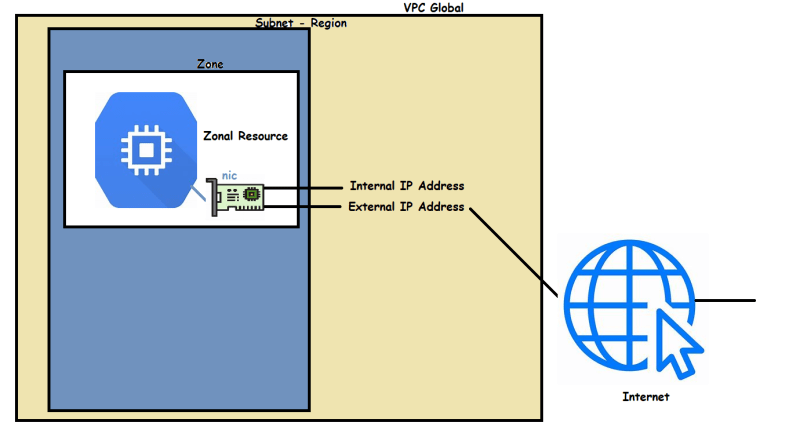




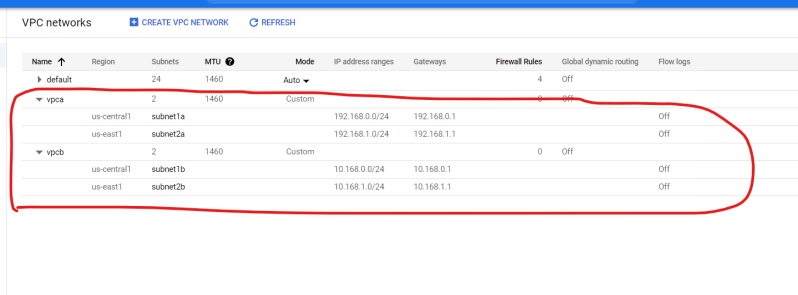
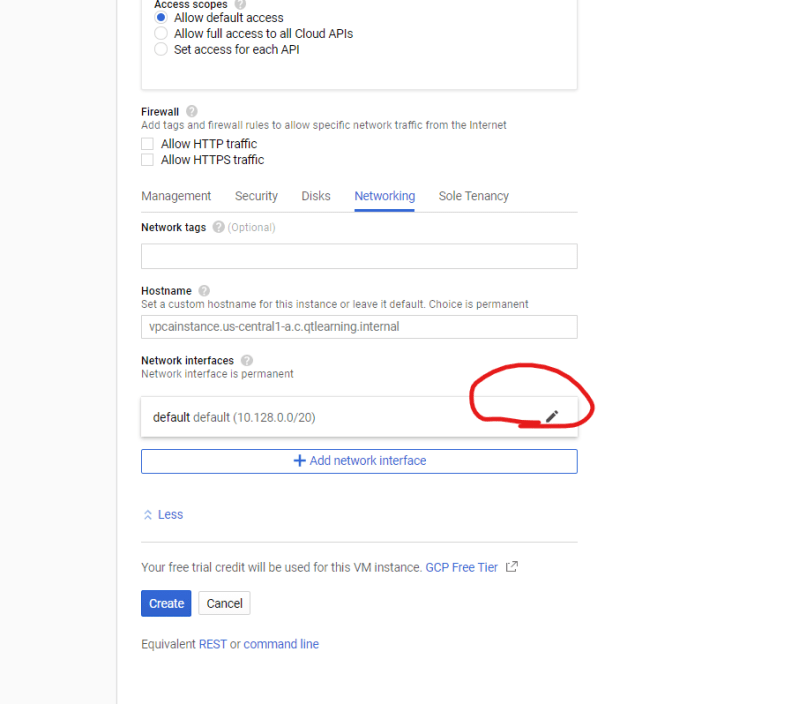
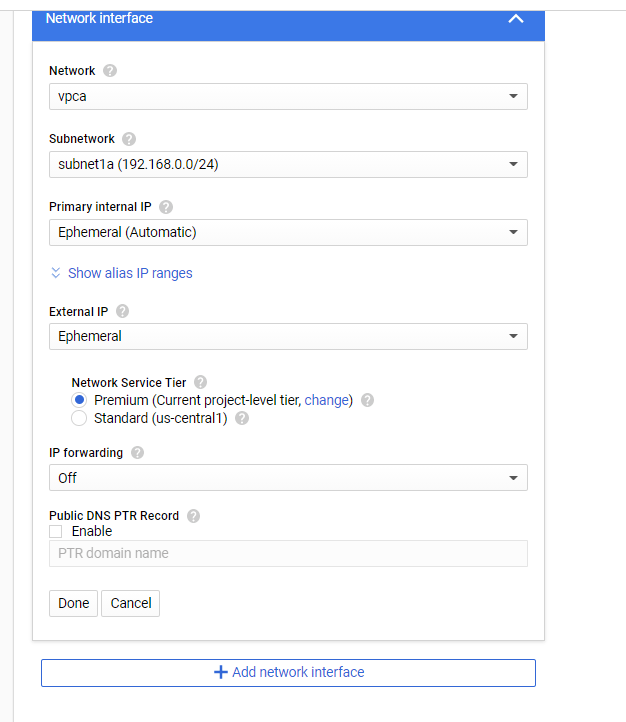
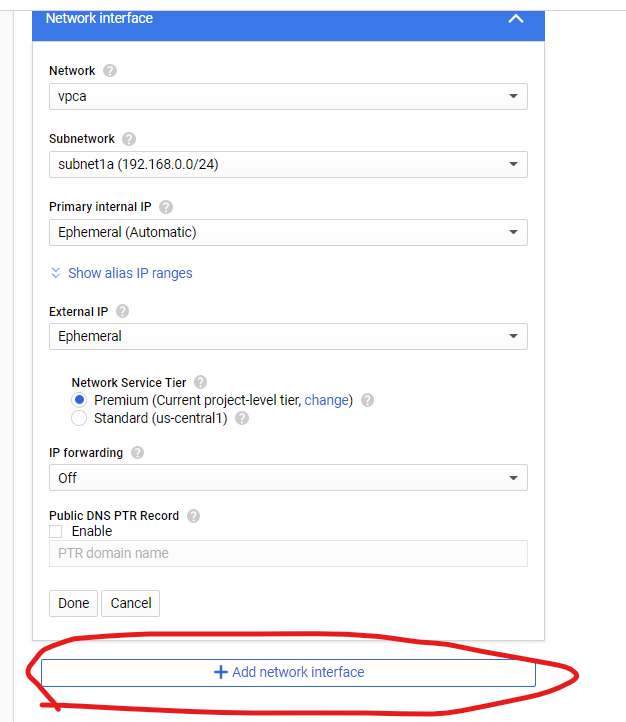
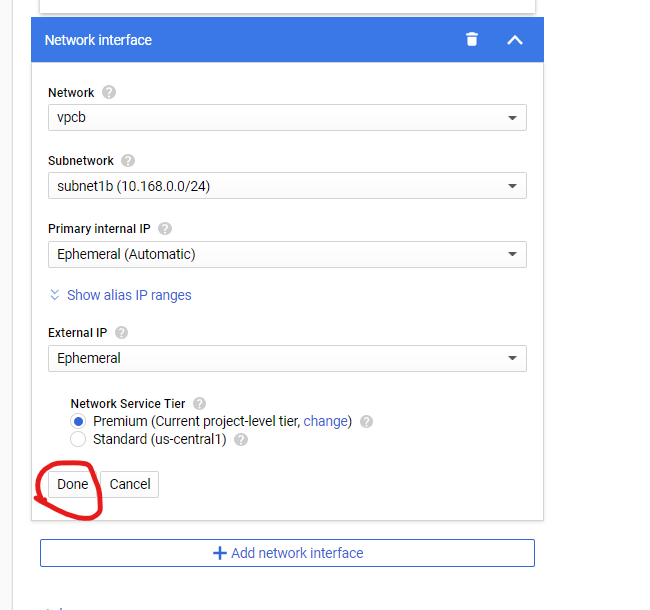
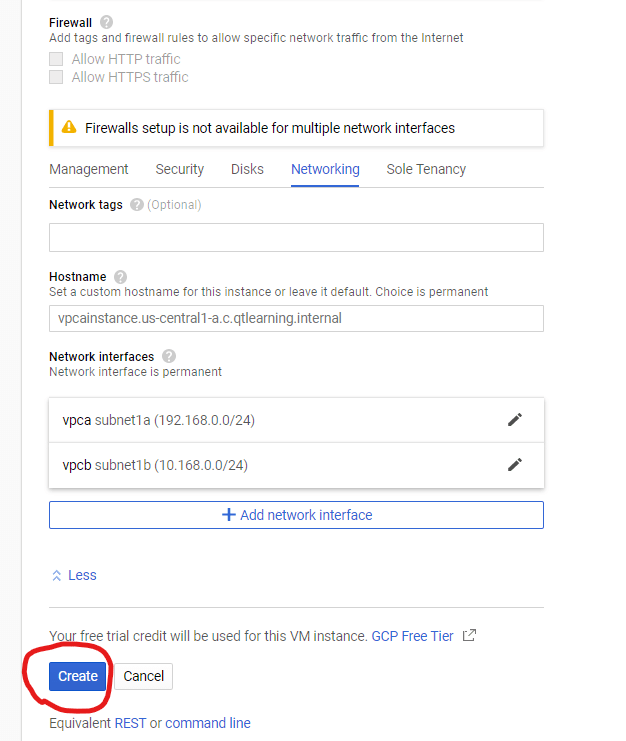
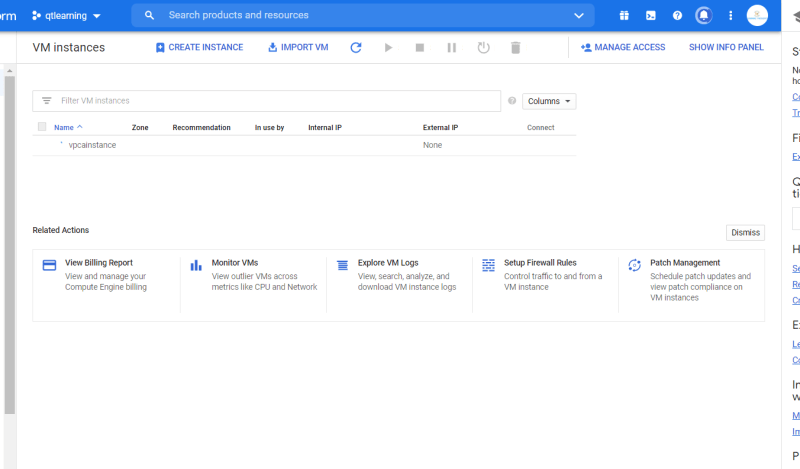
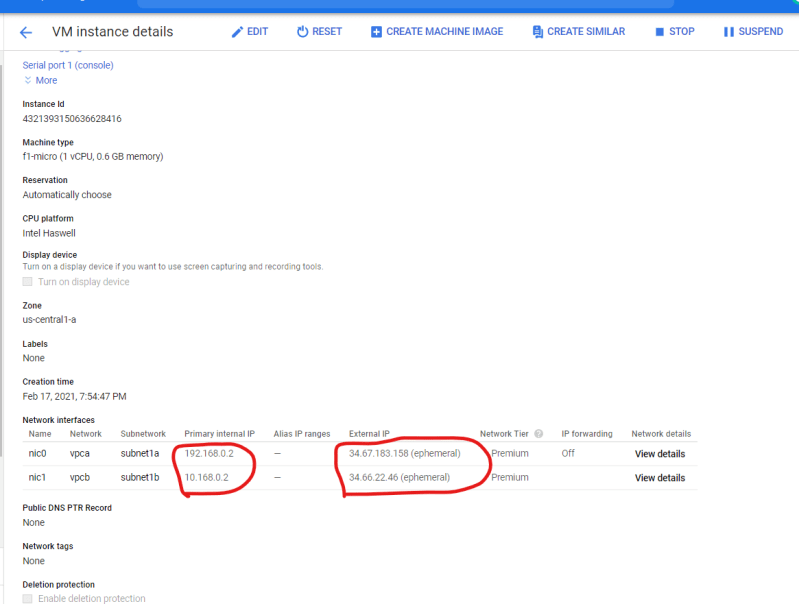




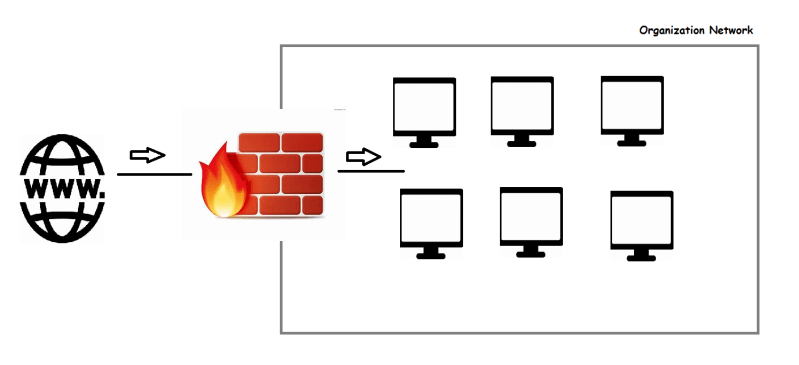
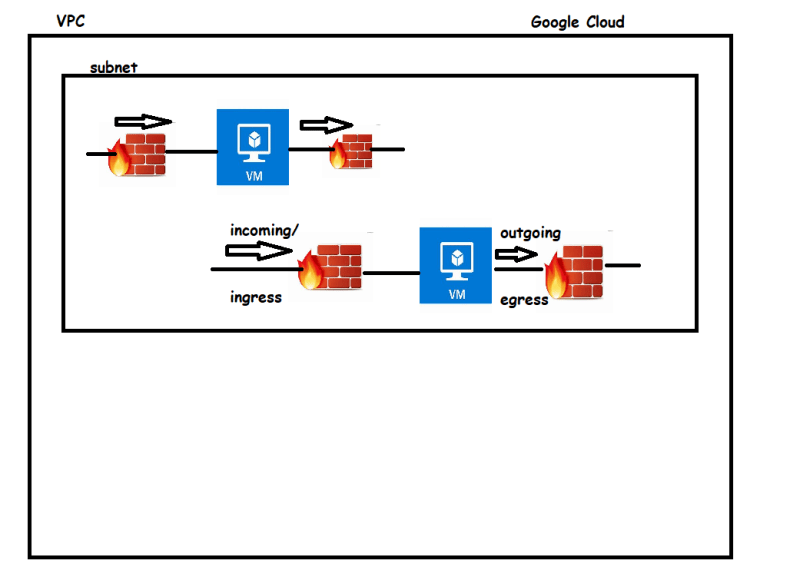
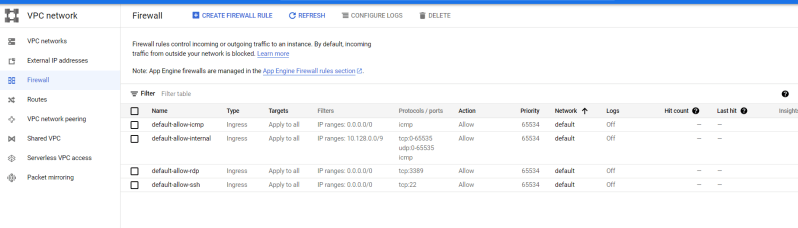
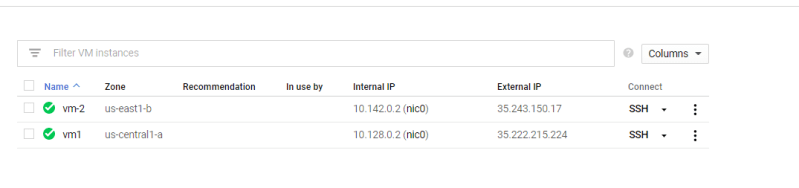
IP Addresses for Resources in GCP

* GCP IP Addressing 
* There are two types of IP Addresses
  + Internal Address
    - Assigned from the subnet range
* IP Addressing can be further classified into
  + Ephemeral: This will still remain even after vm is stopped.
  + Static Ip Address: Can be assigned to any ip in the subnet range

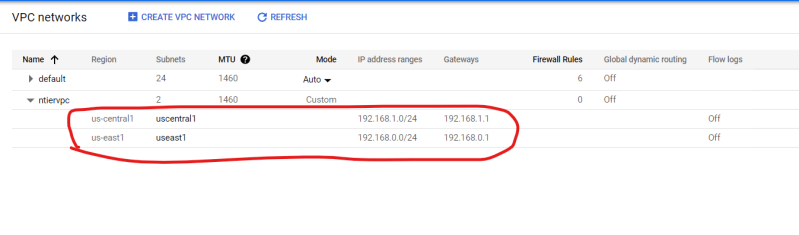
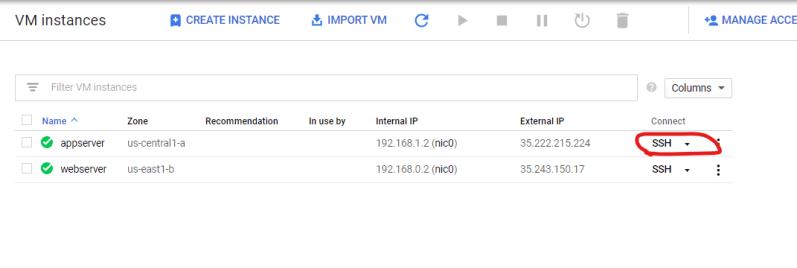
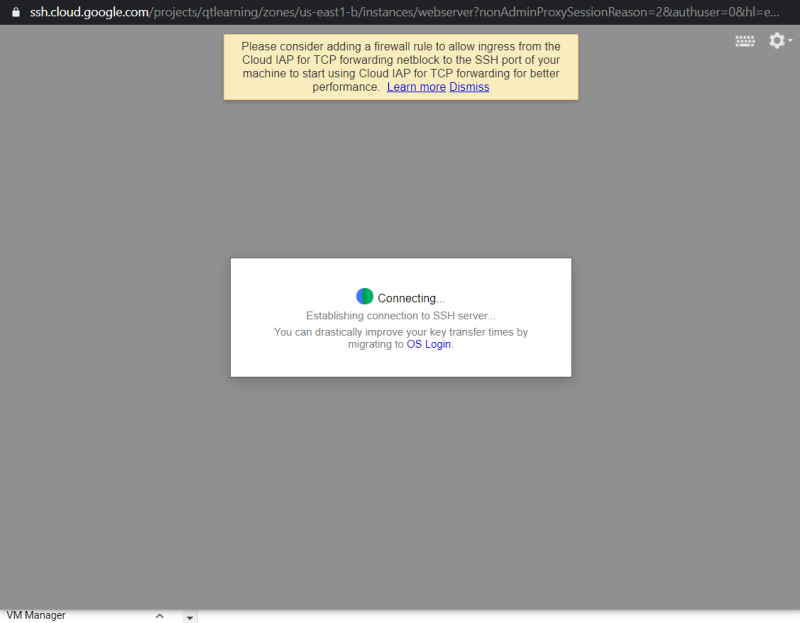
External Address

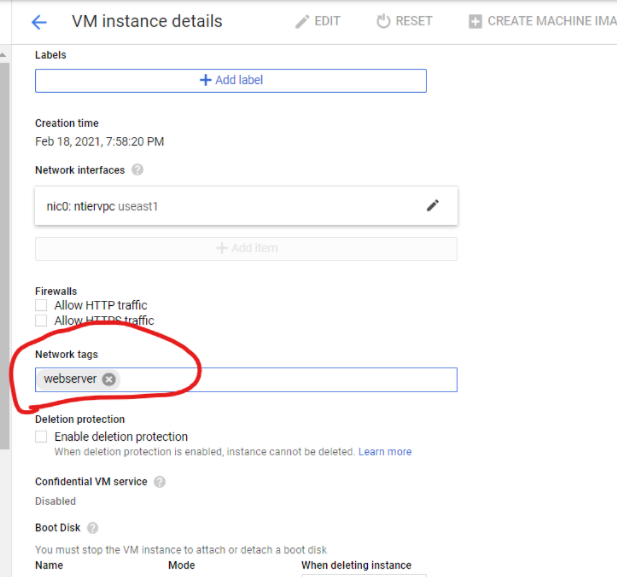
* Assigned to be access from internet
* IP Addressing can be further classified into
  + Ephemeral: This changes when you stop the vm i.e. the ip address is released and when you start the vm you will get a new external ip address
  + Static Ip Address: We can reserve the Static ip and this will remain the same across vm starts and stops.
* Multiple Network Interface Controllers (NIC):
  + VMs can have exactly one internal and external IP per VPC
  + VM can have multiple network interfaces (one per vpc) with external and internal ip addresses
  + Restrictions:
    - 1 NIC per network
    - Two networks cannot have overlapping subnet ranges
    - Maximum of 8 NICs can be assigned to VM
      * less than or equal to 2 VCPUs = 2 NICS
      * grater than 2 CPUs => 1 NIC per CPU
  + NIC’s can be assigned only during vm creation.
* Create a custom vpc with two subnets
  + name: vpca
  + us-central1 : 192.168.0.0/24
  + us-east1: 192.168.1.0/24
* Create a custom vpc with two subnets
  + name: vpcb
  + us-central1 : 10.168.0.0/24
  + us-east1: 10.168.1.0/24
  + us-east1: 10.168.1.0/24 
* Now lets create a compute engine instance in any of the vpc’s  
* We can continue instance creation but lets add one more network interface from different vpc     

Firewall

* Firewall is all about allowing and denying network traffic
* Corporate Firewall/Tradational firewall 
* Google also has a virtual firewall which is defined at network level but is enforced for each instance 
* No firewall means
  + no ingress => no packets are allowed to communicate into GCP VM Instance
  + full egress => all the packets will be allowed to communicate from GCP VM Instance
* Firewall has Rules, which managed external & internal access to resources
* Implied => deny all ingress
* Implied => Allow all egress
* Firewal Rule Components
  + Direction: Ingress or Egress
  + Target: GCP resources the rule applies to: Entire network, Target Tags, Service Account
  + Source/Destination Filter: Incoming Sources, Outgoing Destination that the rule applies to
  + Action: Allow or Deny
* The default firewall rules created by GCP for default vpc are as shown below 
* Let’s quickly create a virtual instance in default network with any linux os in us-central1
* Then create one more virtual machine in default network with any linux os in us-east1 
* Now connect to vm1 using browser ssh session
* Vm1 which we created is able to ping internet and also vm2 using internal ip

Exercise

* Create a custom vpc with two subnets in us-central1 and us-east1 
* Now create a vm in us-central1 and us-east1 
* Try to connect to the vm created. 
* We will not be able to login as default/implied firewall rule comes into play
* Now create a firewall rule which allows ssh traffic for the vm with tag webserver



* After this we should be able to login into web server