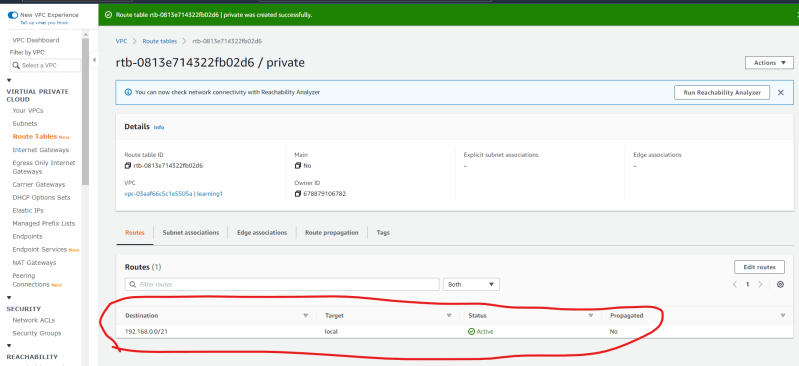
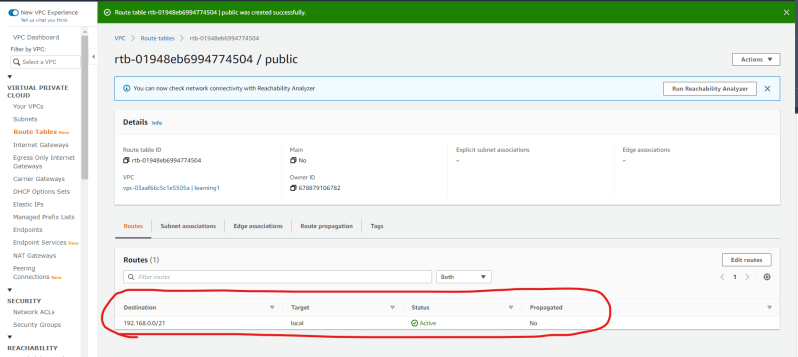
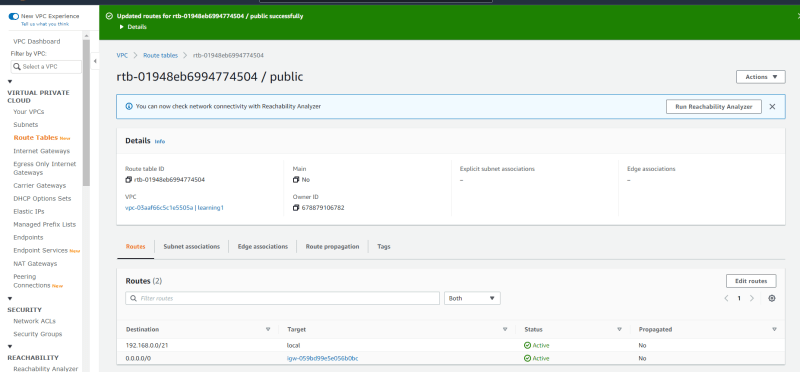
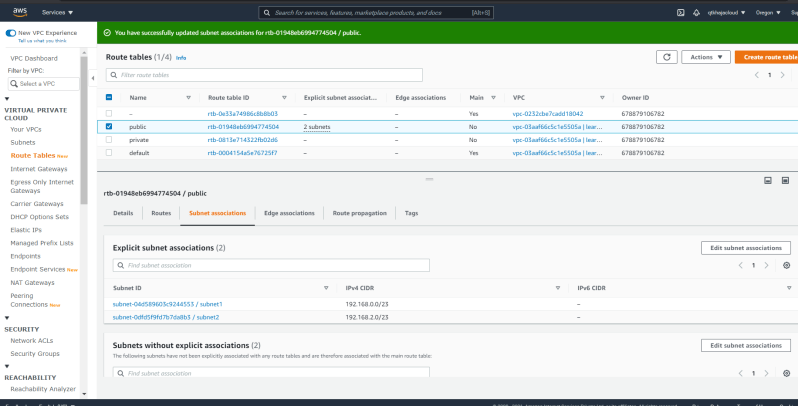
**Public and Private Subnets**

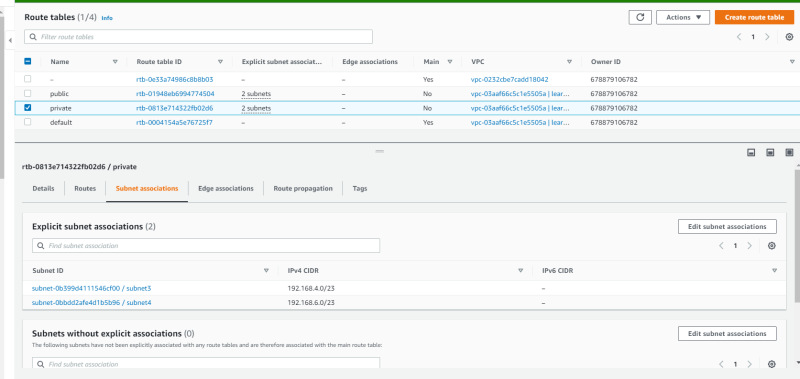
* So far, we have created a network and internet gateway, we have created a route from default route table to internet gateway
* So, AWS allows us to create route tables in vpc.
* Subnets can be associated with route tables
* Let’s create two route tables 

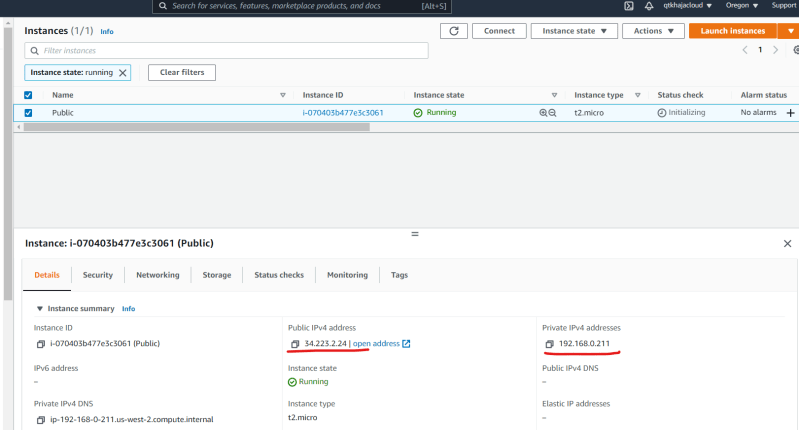
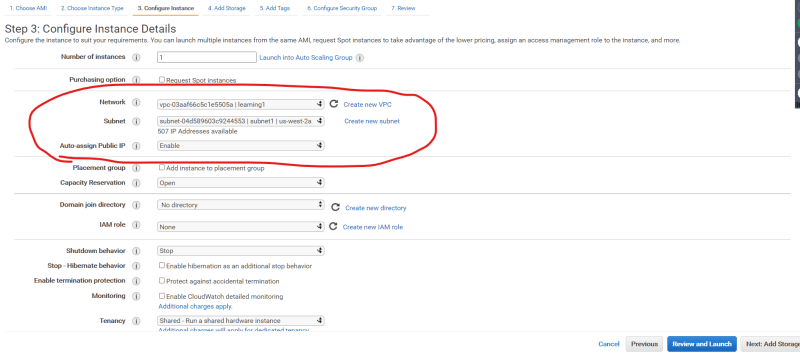
Route table which has route to internet gateway is public. 

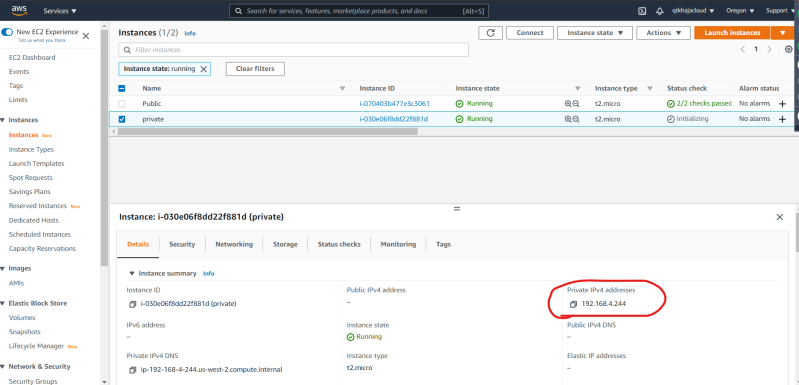
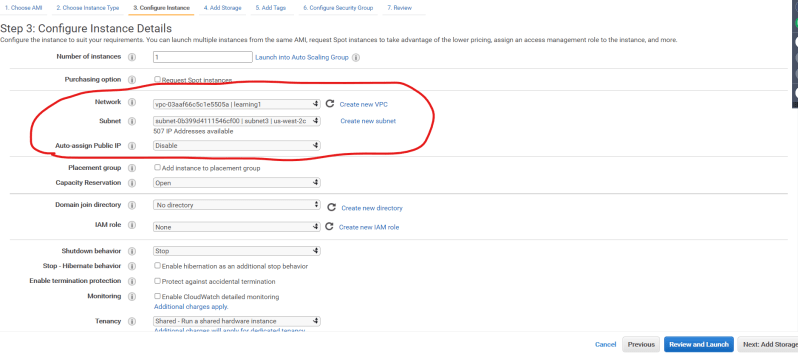
If we have not associated subnets to a specific route table, then by default they will use default route table (main yes)

This can be changed when we add explicit associations to the route table





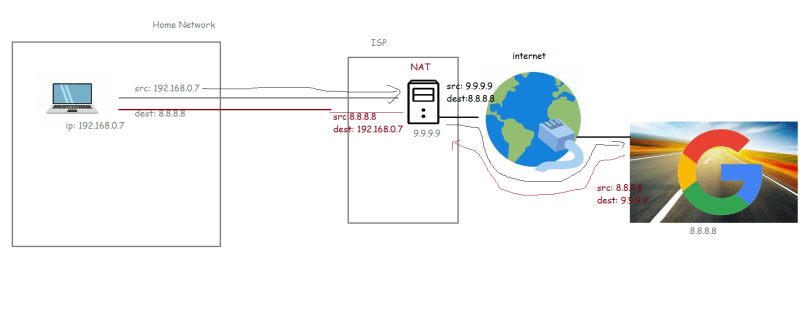
* A subnet is considered as private if it does not have route to internet gateway and is considered as public if it has route to internet gateway. So in our case
  + subnet1 and subnet 2 are public
  + subnet3 and subnet 4 are private
* Let’s create one ec2 instance in public subnet with public ip address 

one ec2 instance in private subnet with only private ip address Let’s ssh into ec2 instance in public subnet 

We cannot directly access ec2 in private subnet, we can use ec2 in public subnet to connect to ec2 in private subnet (bastion host/jump host)

* Now let’s see if the machine in private subnet gets internet access (no)
* In many cases, we would require machines in private subnet to access internet for security updates, software installations etc. But we don’t want machines in private subnet to be accessed from internet in any case.

**Network Address Translation**

* Basic NAT 
* Now if we can somehow implement NAT in our VPC then our private subnets can access internet 