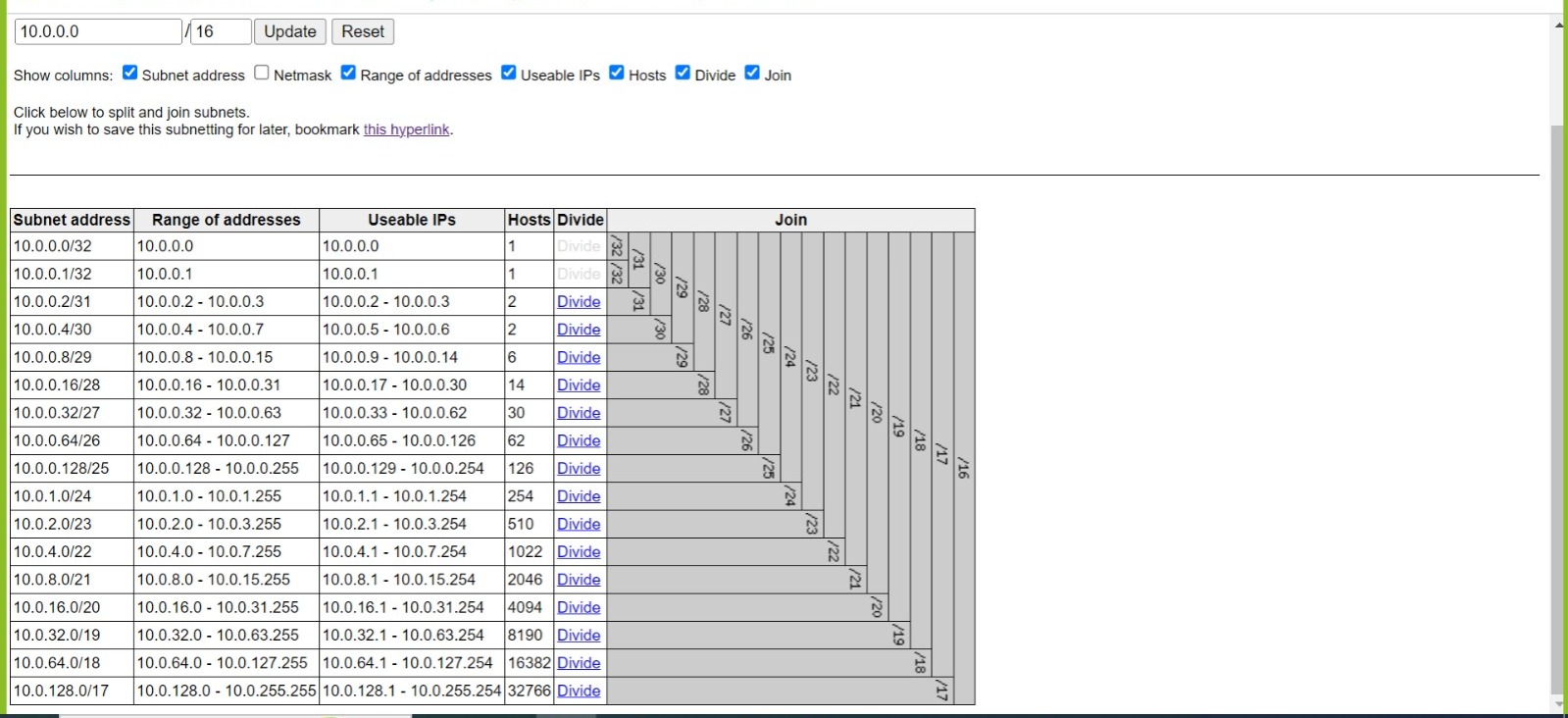
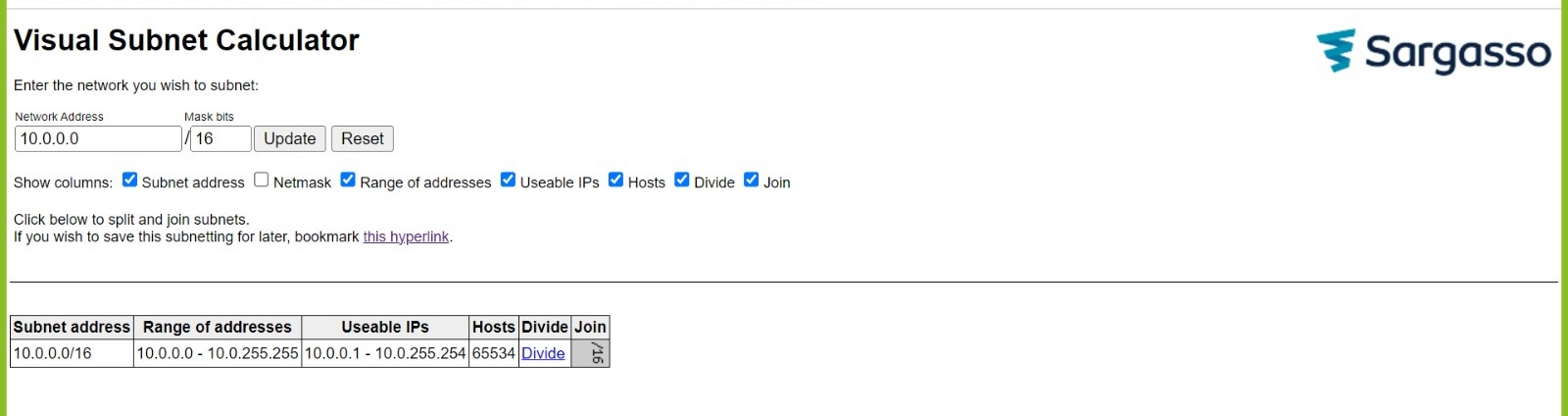
How we can calculate /16 CIDR into smaller CIDR’s (that is /20)



<https://www.davidc.net/sites/default/subnets/subnets.html>

This is the site from where we can easily calculate different networks of a CIDR.

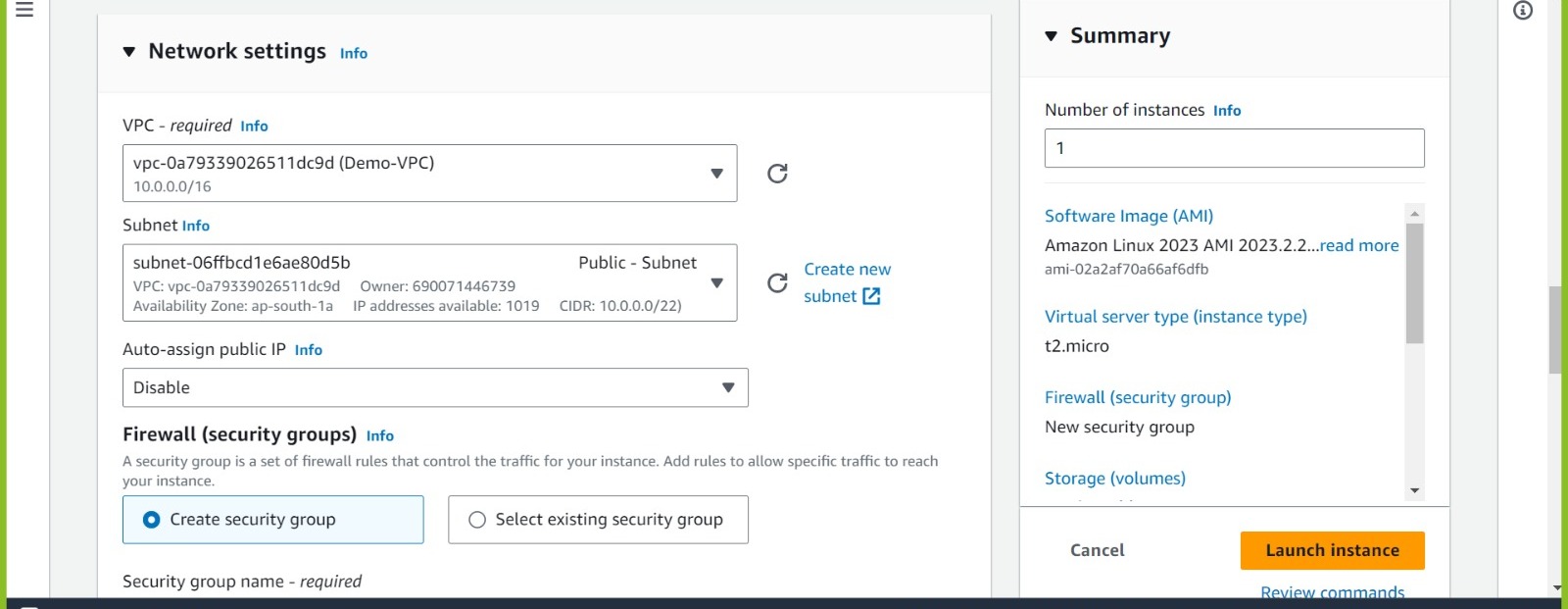
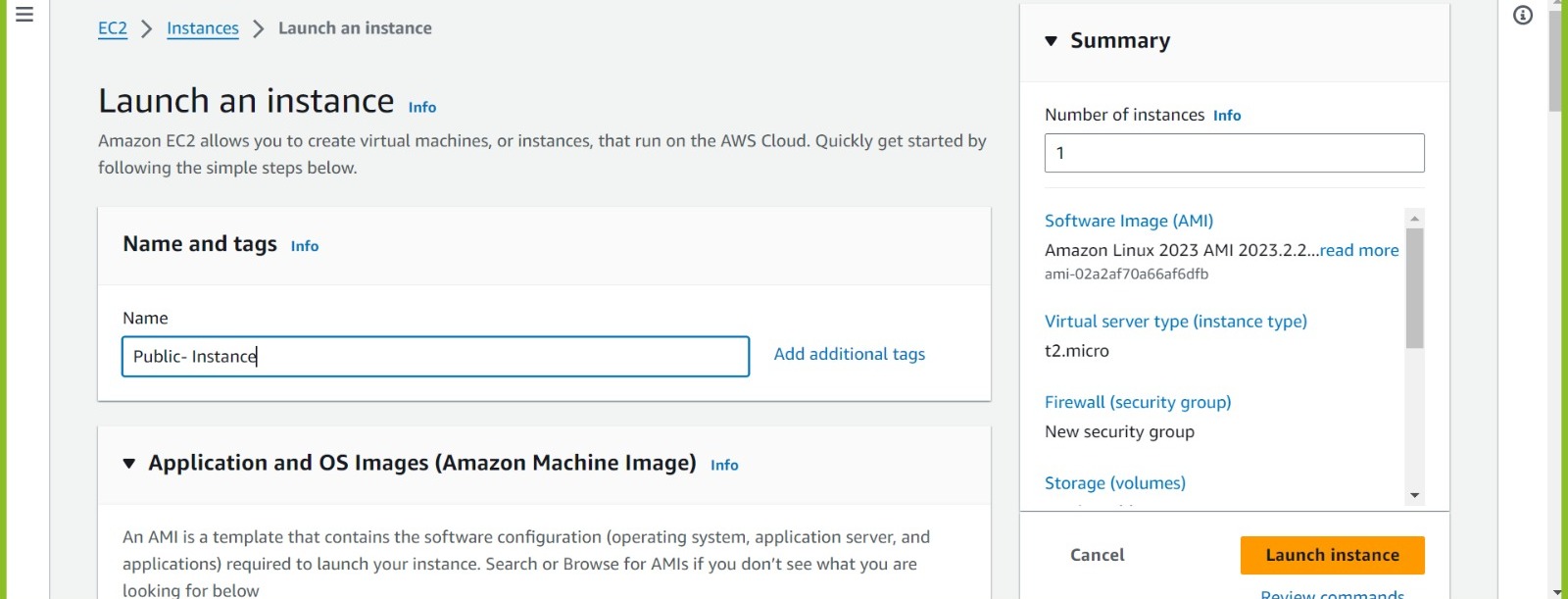
On the top write the range like (10.0.0.0/16) and click on update.



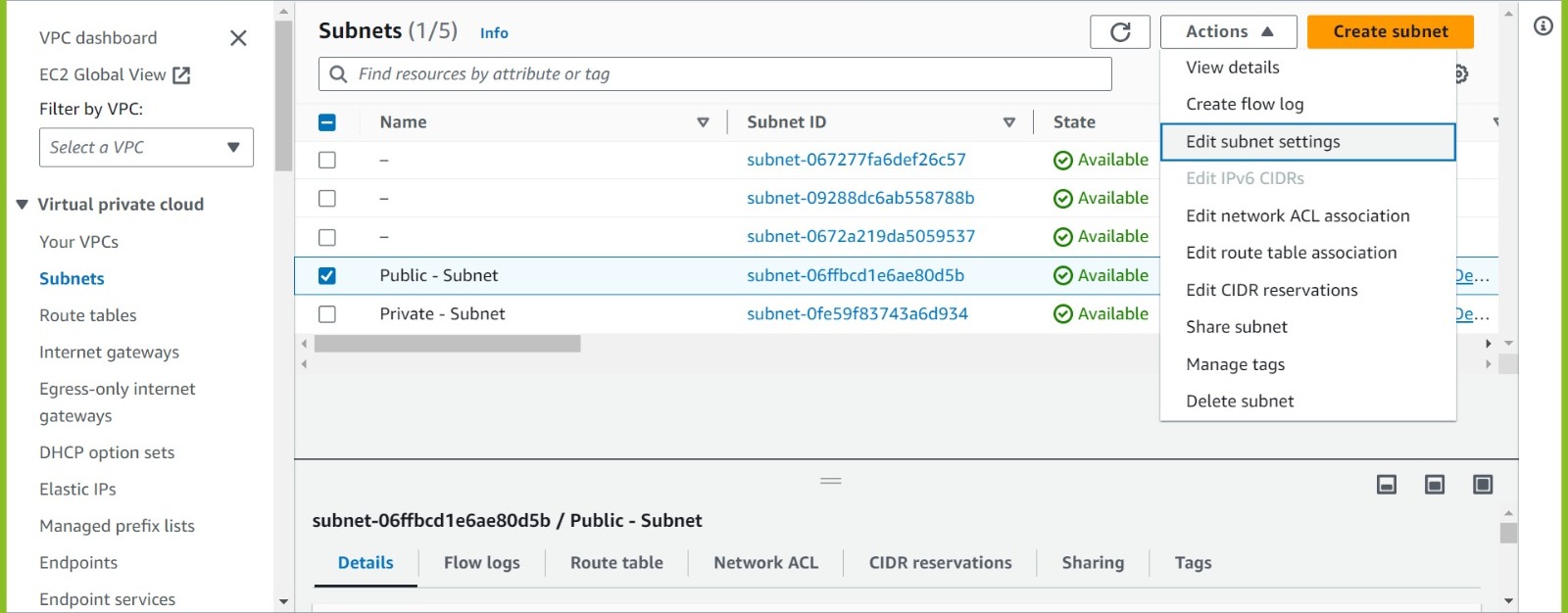
Now start dividing this network and you will get multiple network range till we get the minimum IP’s.

As we have successfully created VPC in our last lecture so now create EC2 into the subnets (into our VPC network).

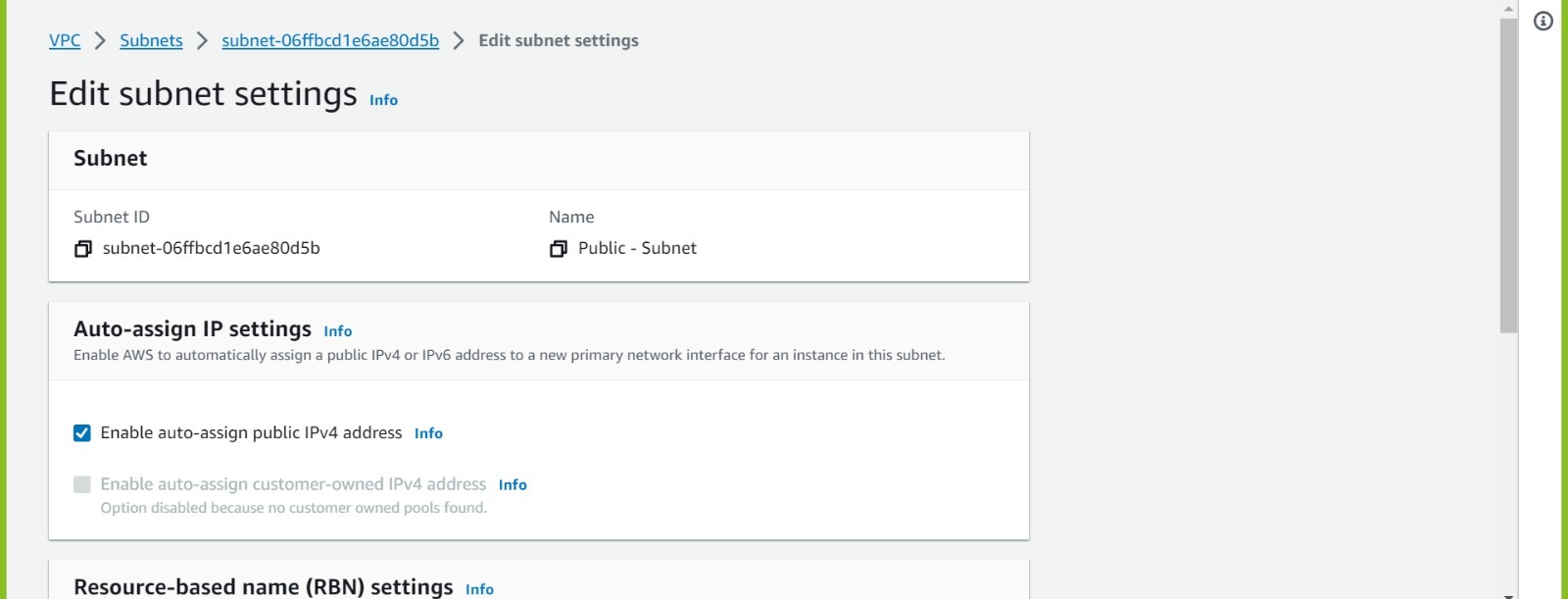
## Now Launch Public Instance



* OS (windows)--->instance type (t2.micr0) ---> keypair(create new or keep old one) ---> security group (new or existing)-->storage(gp3)--->launch
* As you can see if we select Demo-VPC then automatically subnet will get selected but auto-assigned public IP is disabled.
* As we are using public instance, we required public IP so we need to change it from disable to enable.
* But to do this process automatically as **enable** go to public subnet settings.

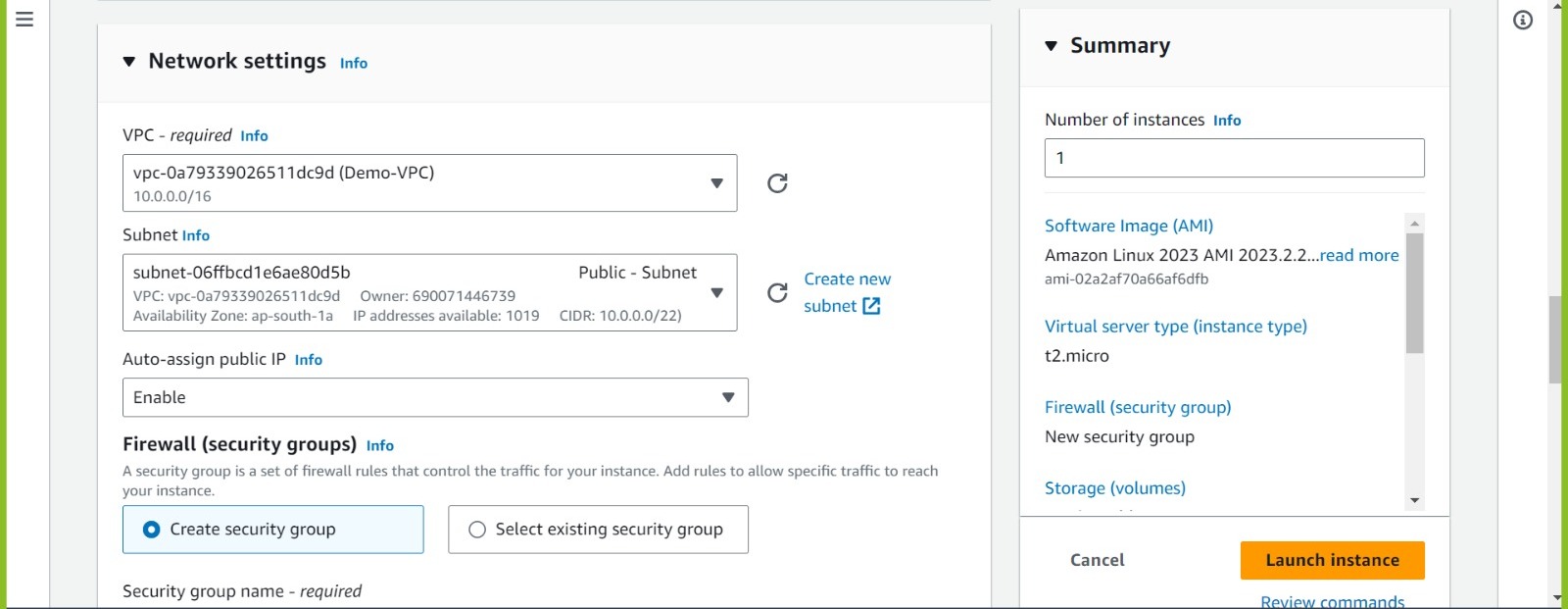


subnets--> action-->edit subnet settings

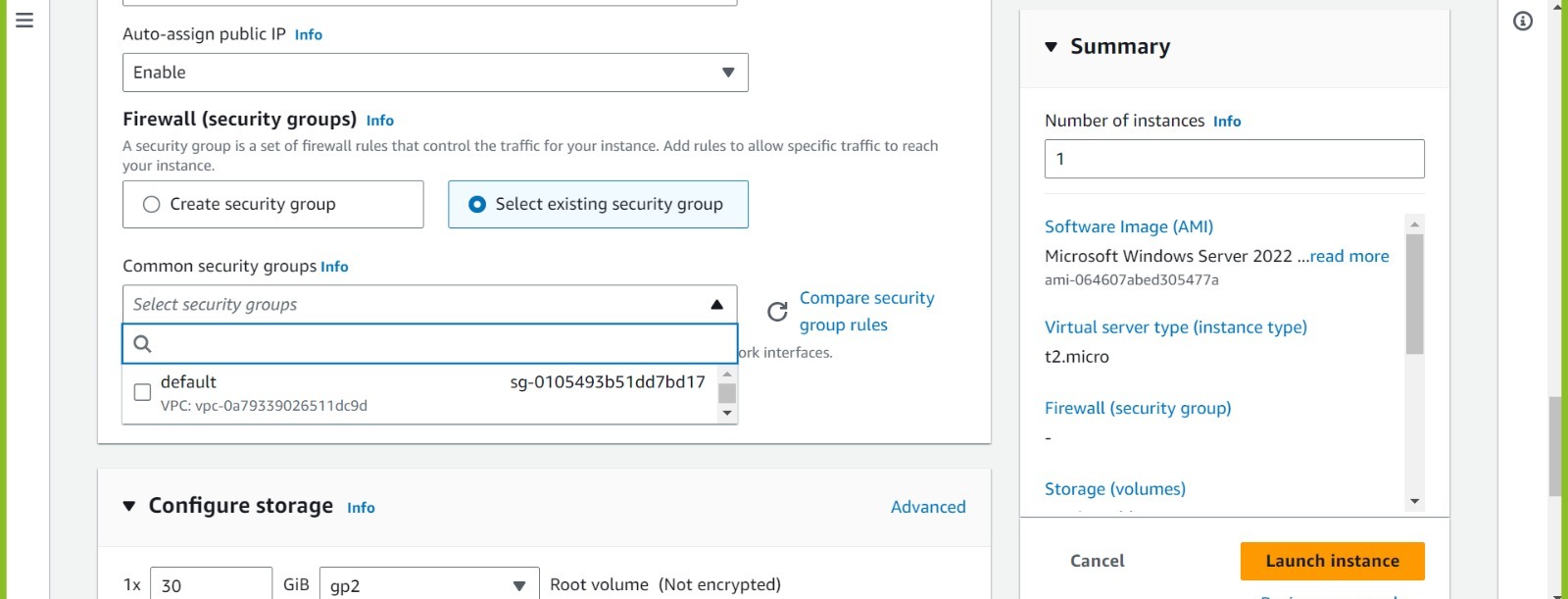


Enable it and save the changes.

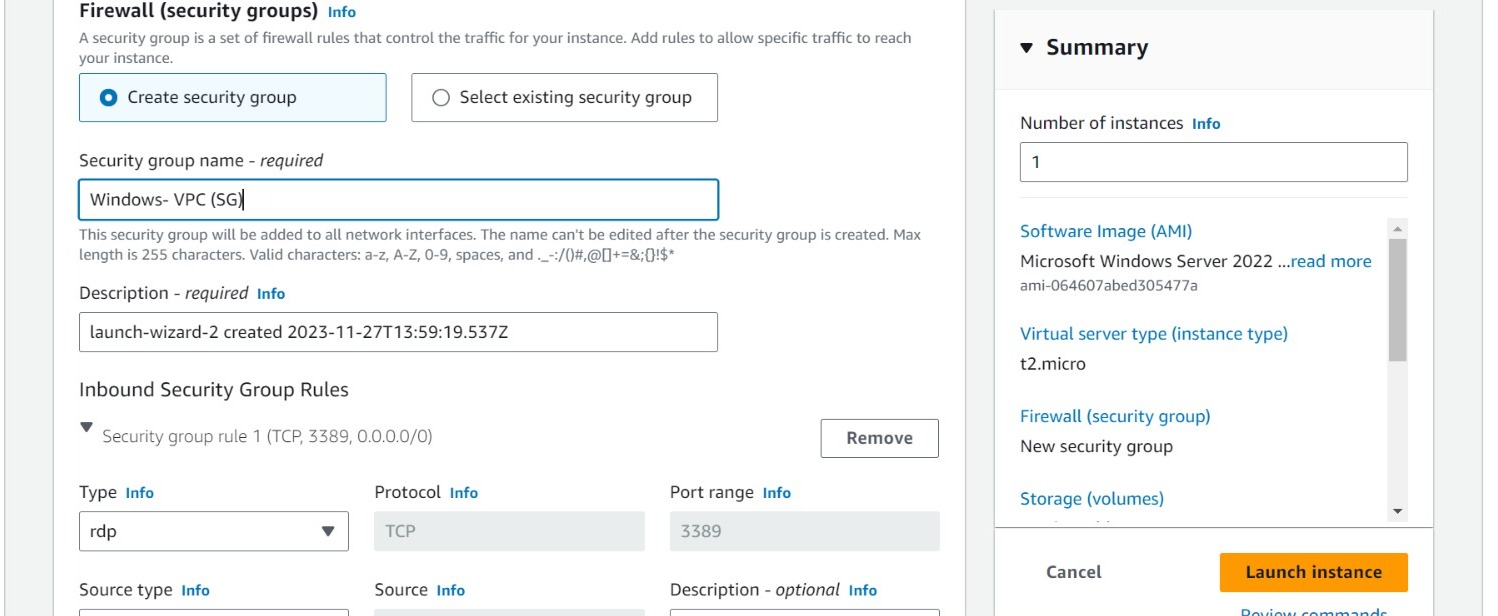
Now when while launching instance when we select public subnet then auto assign public IP will be automatically enabled.

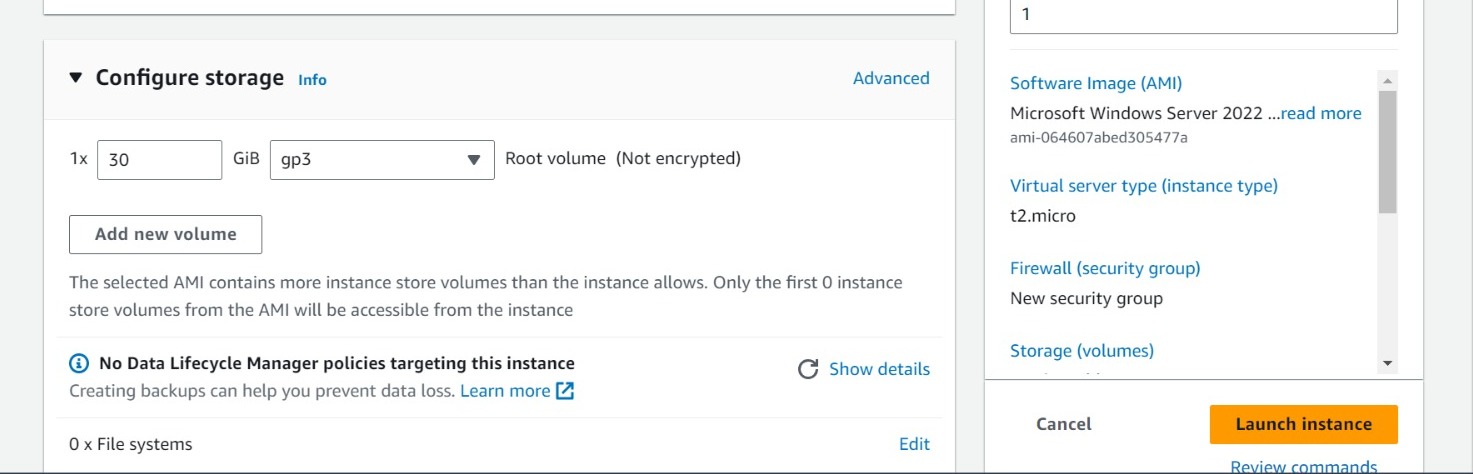


Refresh it and again if you select it will automatically shows enable.

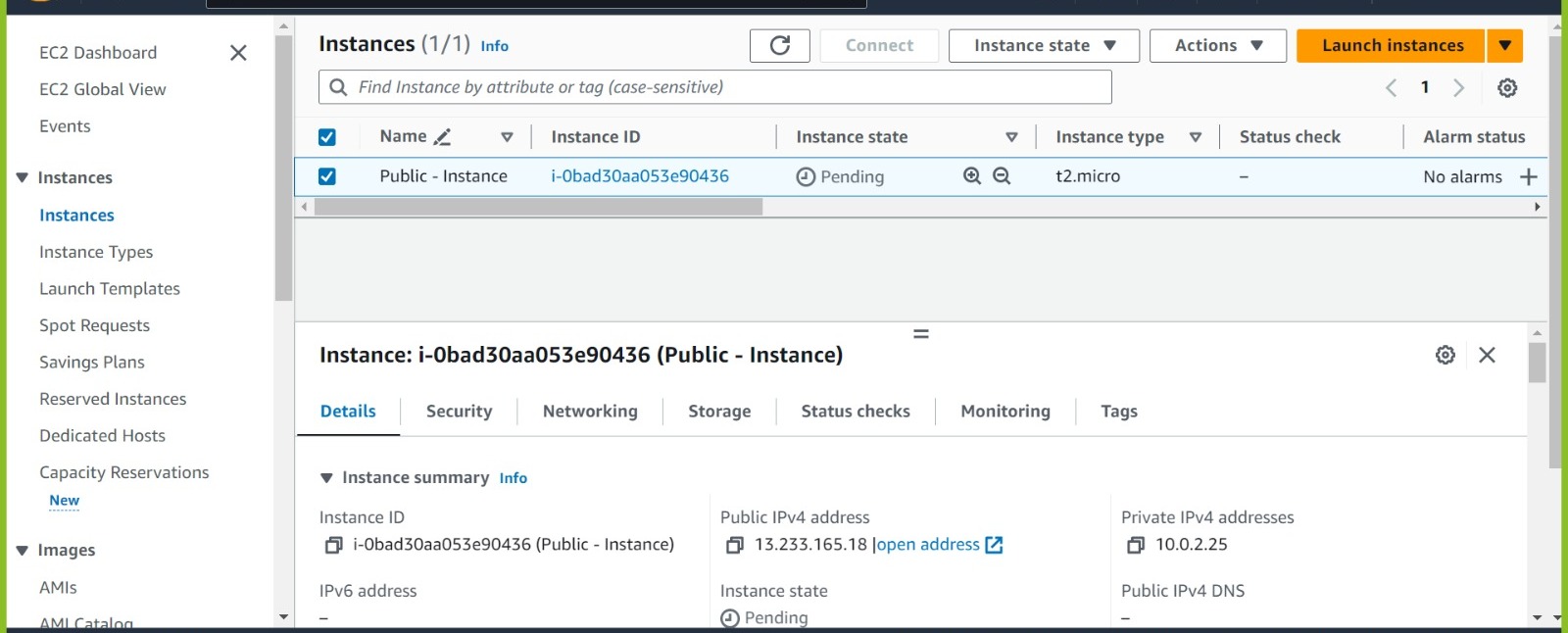


As SG are VPC based in this VPC we have not created any SG yet so we need to create it.



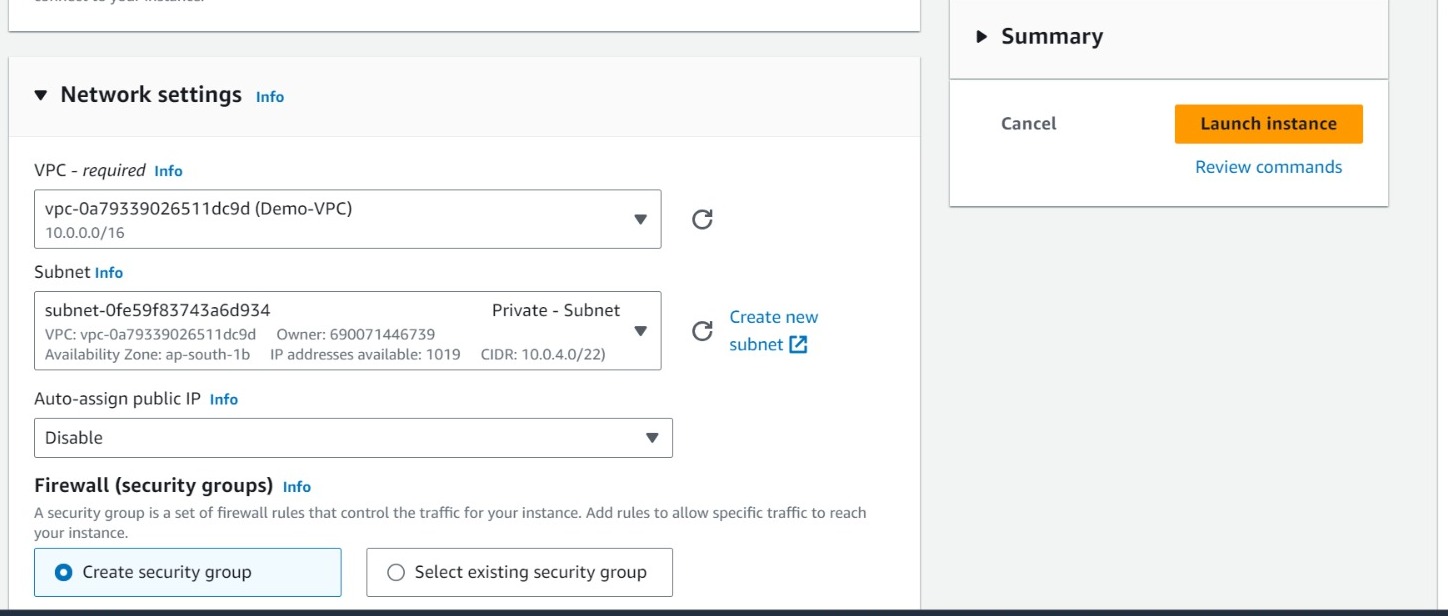


Change the volume and launch instance.

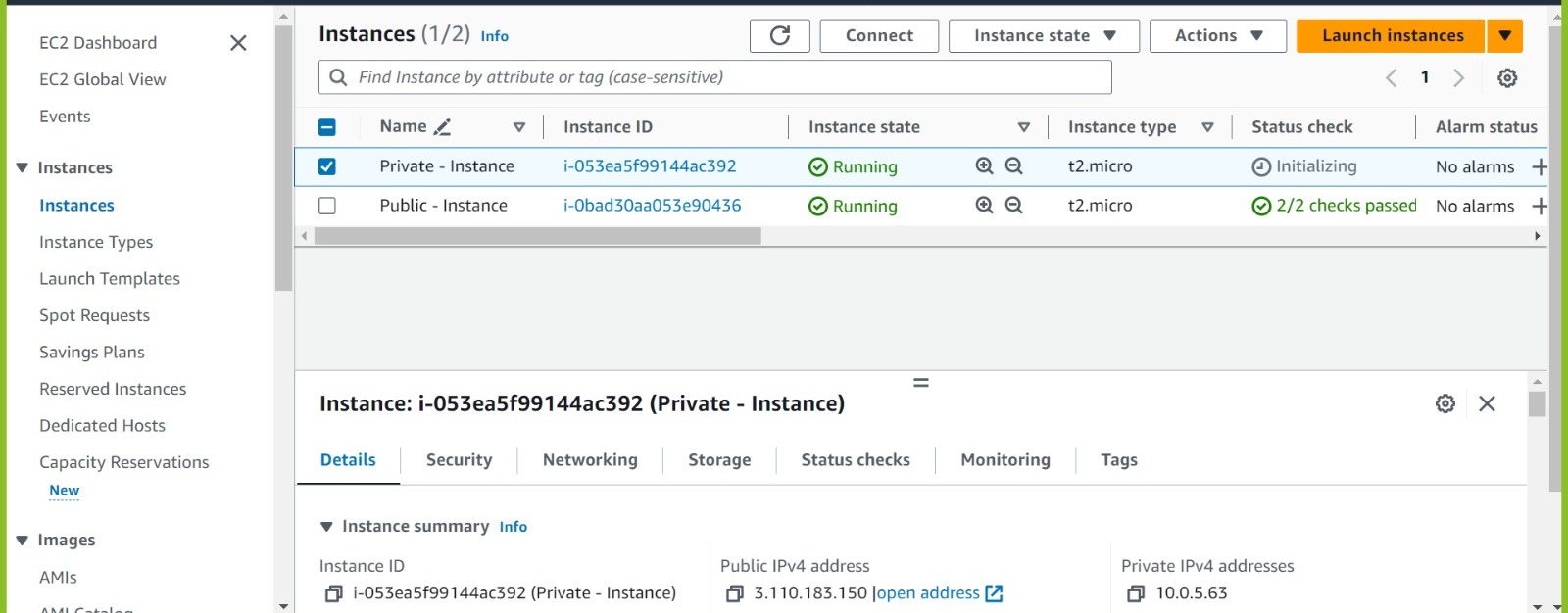


Instance launched successfully

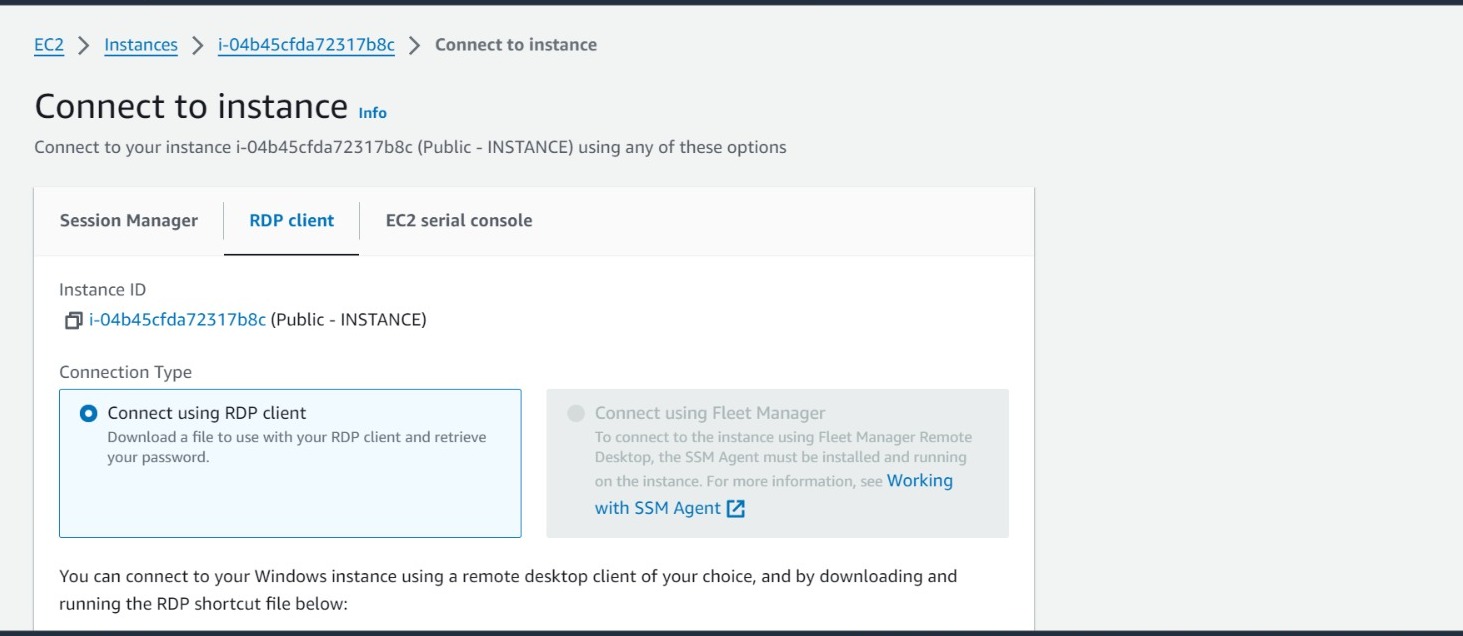
## Now launch Private Instance



* If we select private subnet then automatically, we will get auto-assigned ip as disable .
* But in this case, we will keep it as enable to check we can connect to internet or not.
* As this instance is connected to private subnet, we don’t need public Ip but here we enabled for testing purpose.

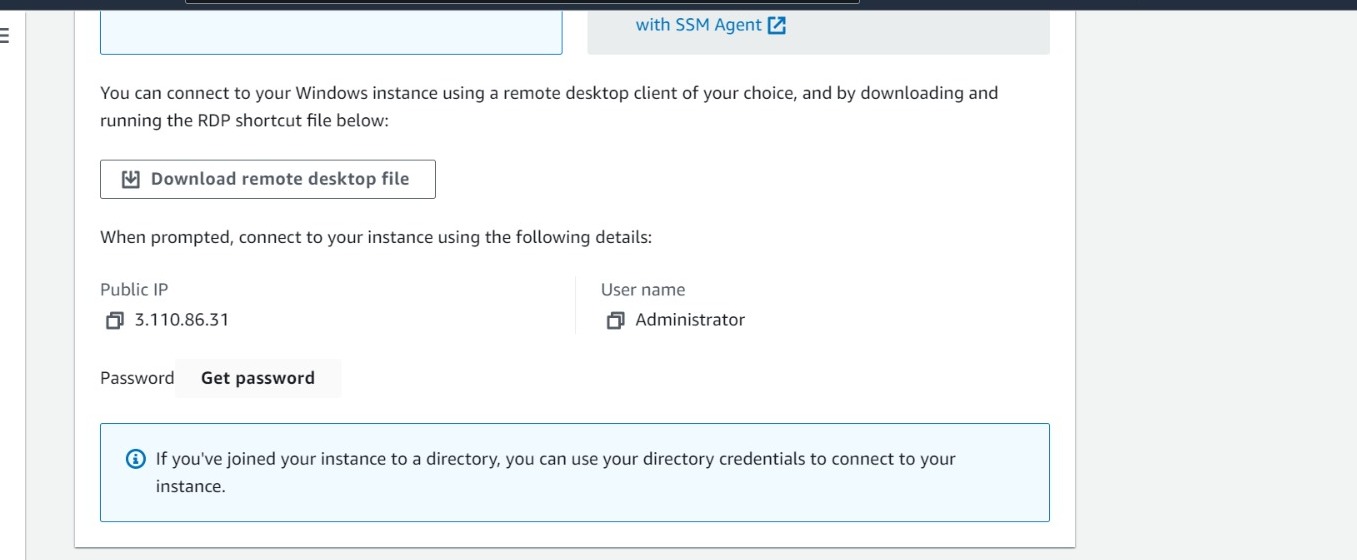


Private Instance launched successfully.



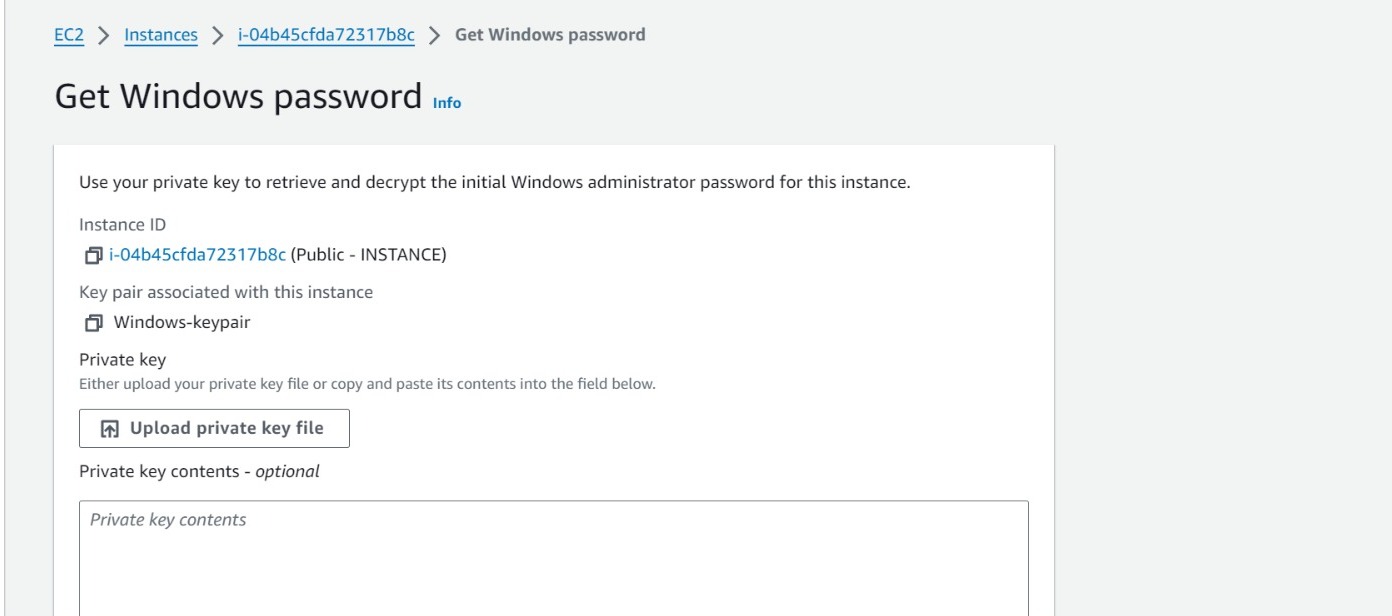
Now Public - Instance to server

First download the Remote desktop

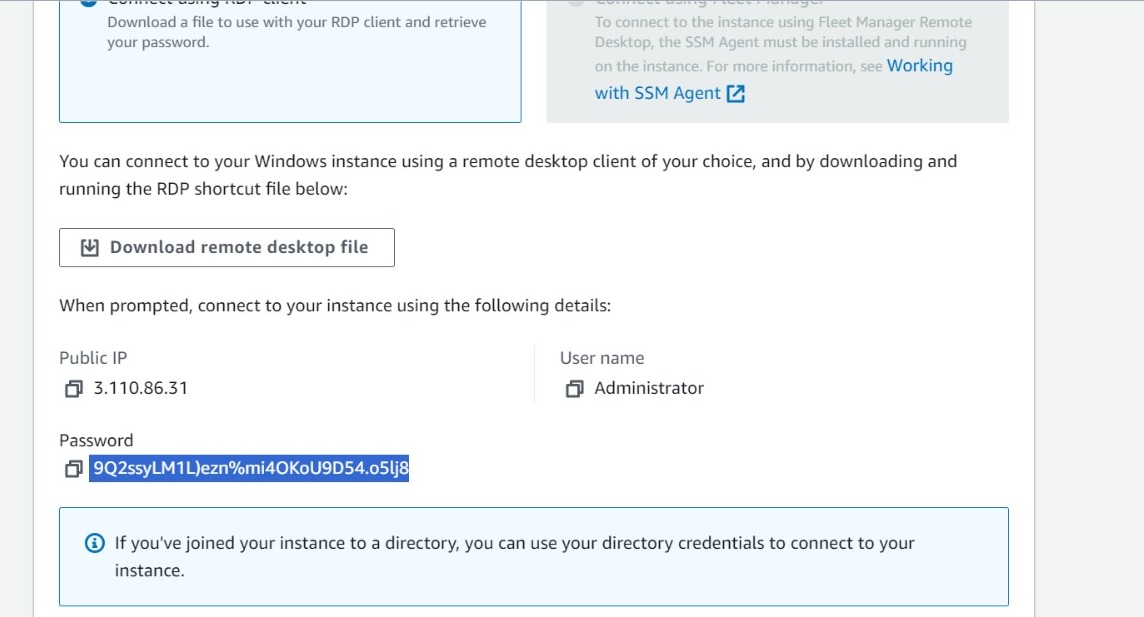


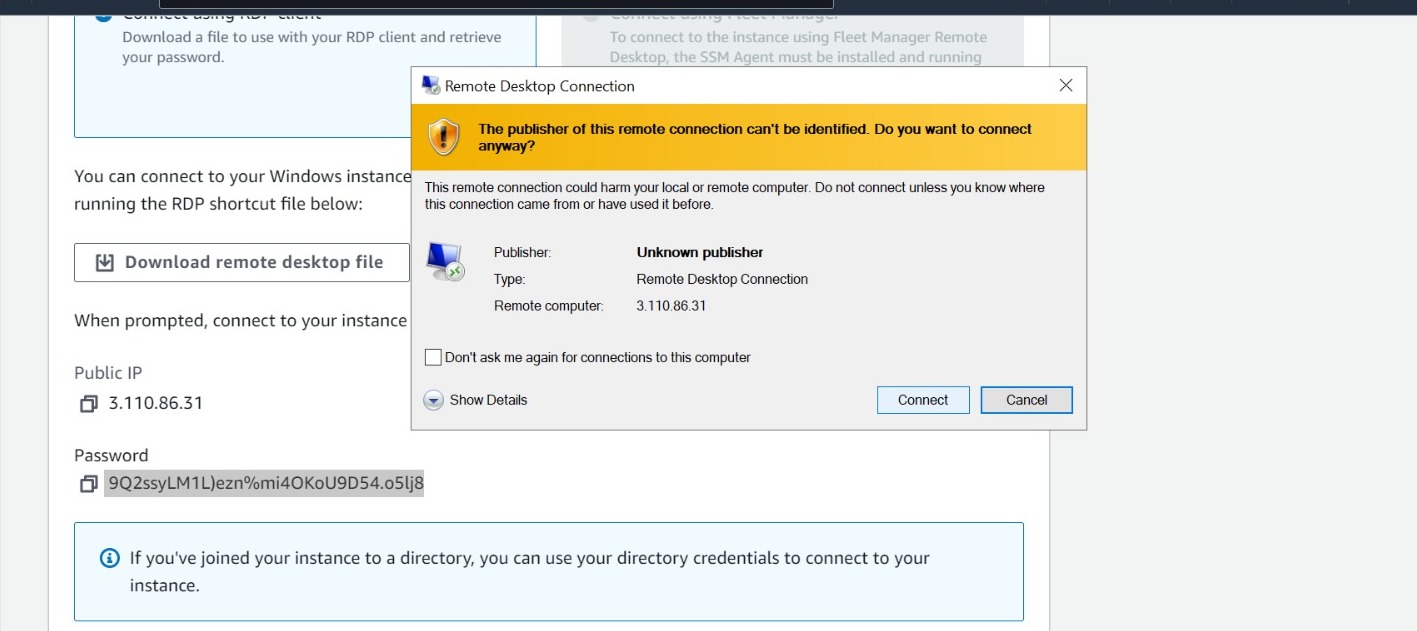
Then click on Get Password

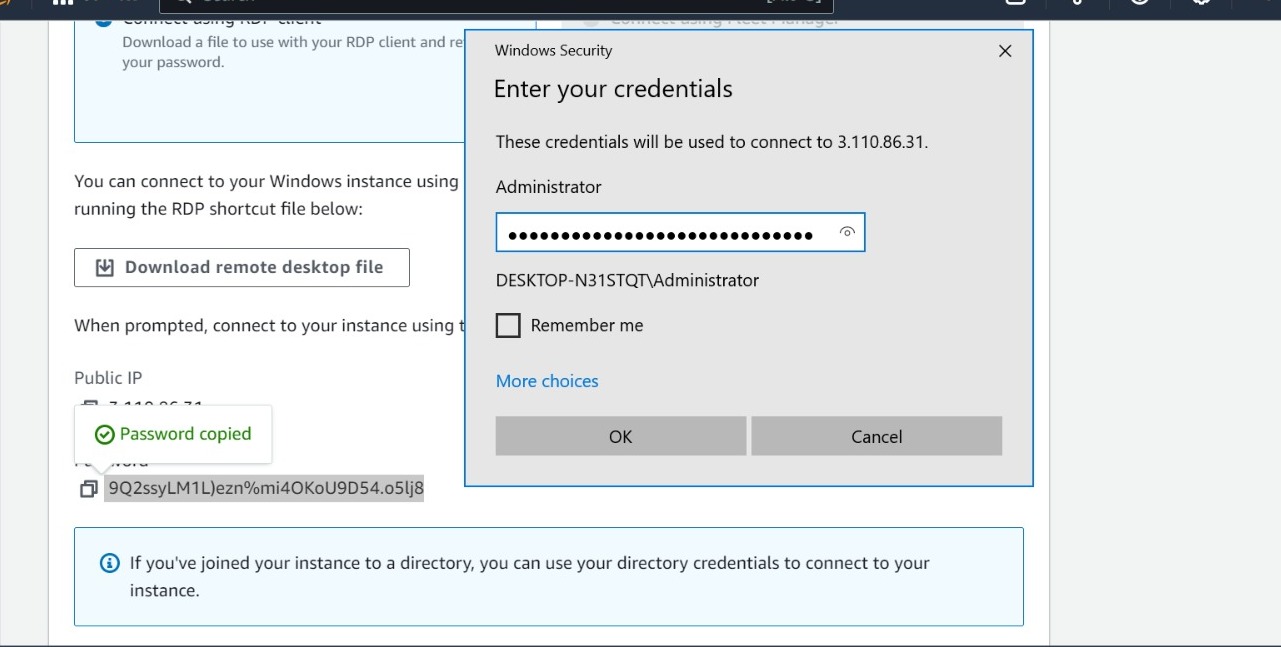
upload key pair and get the password.



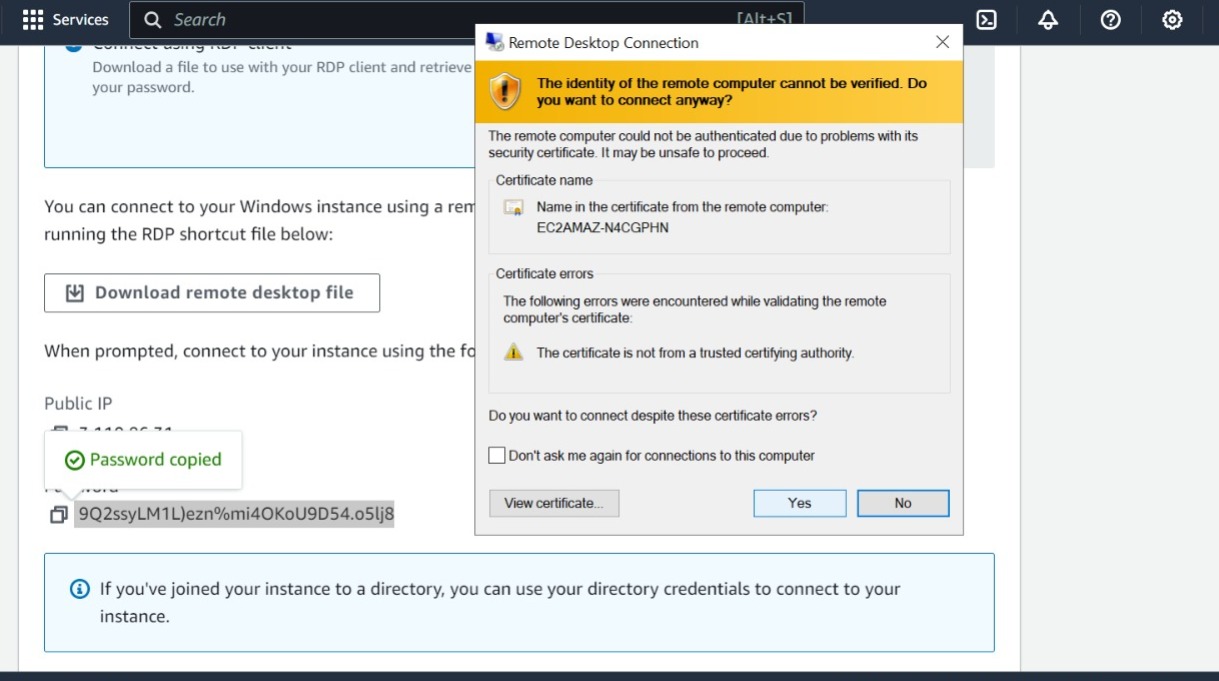
And get decrypted password.



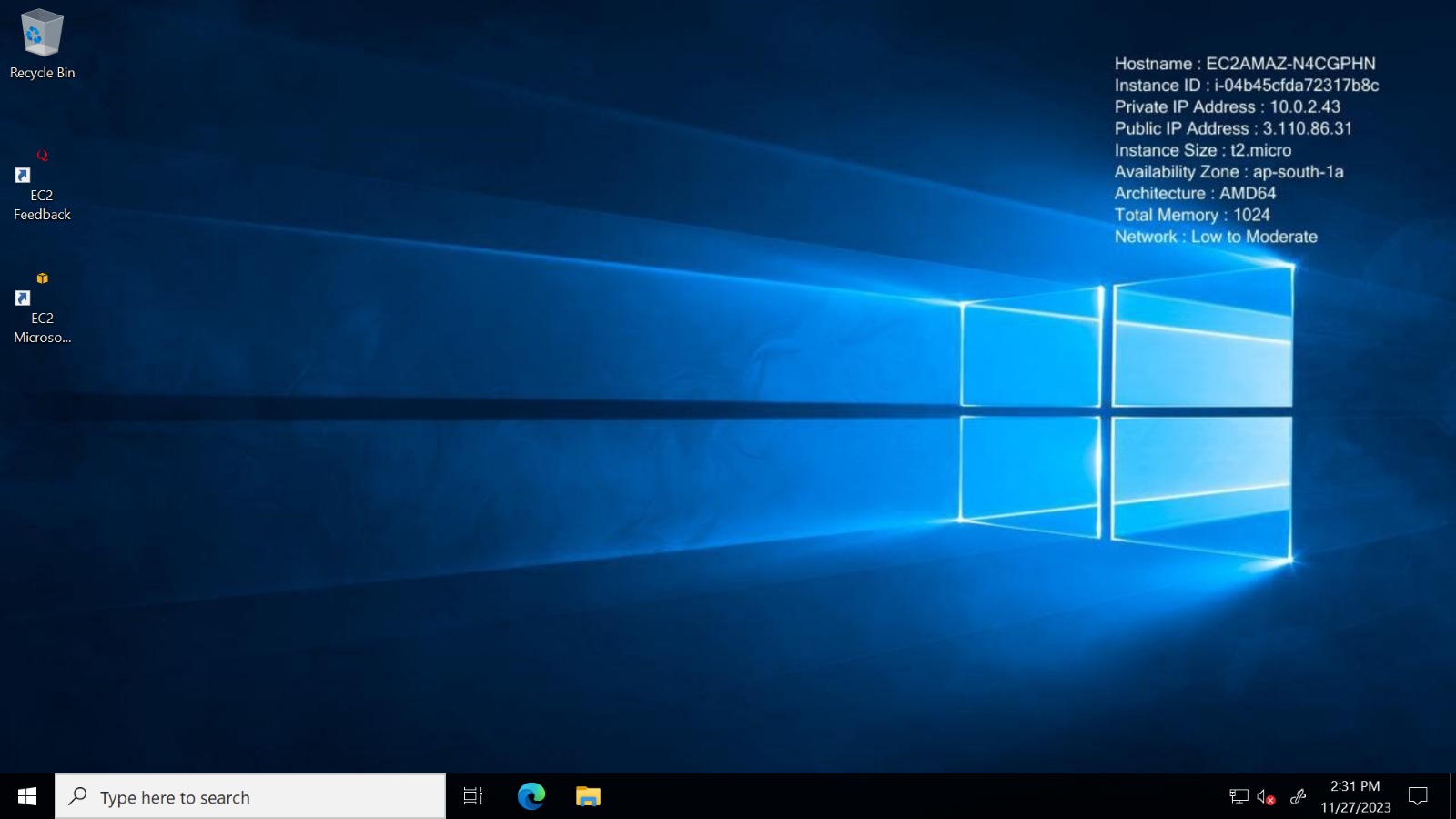


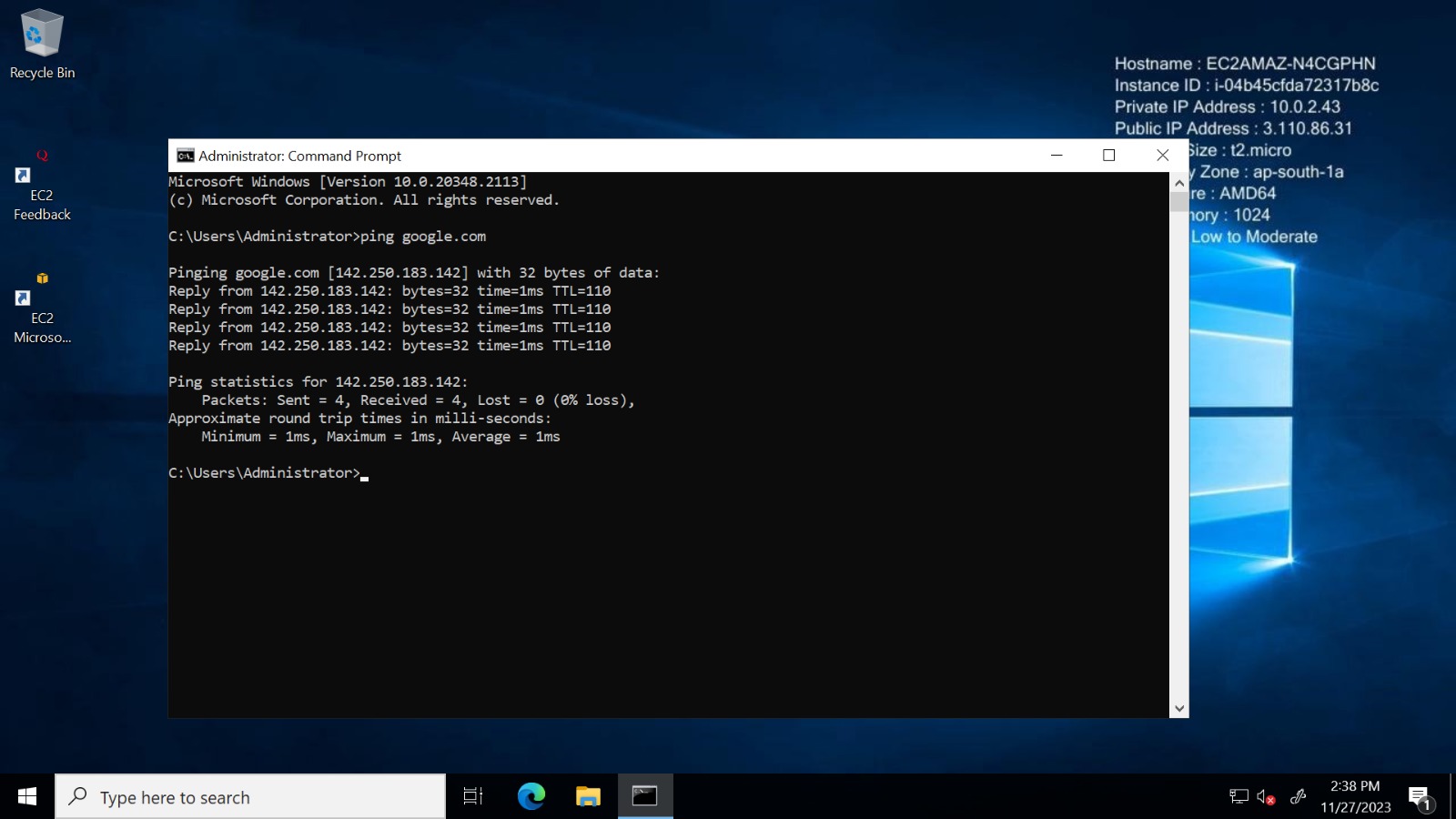


Paste the password which we have created.



Click on yes

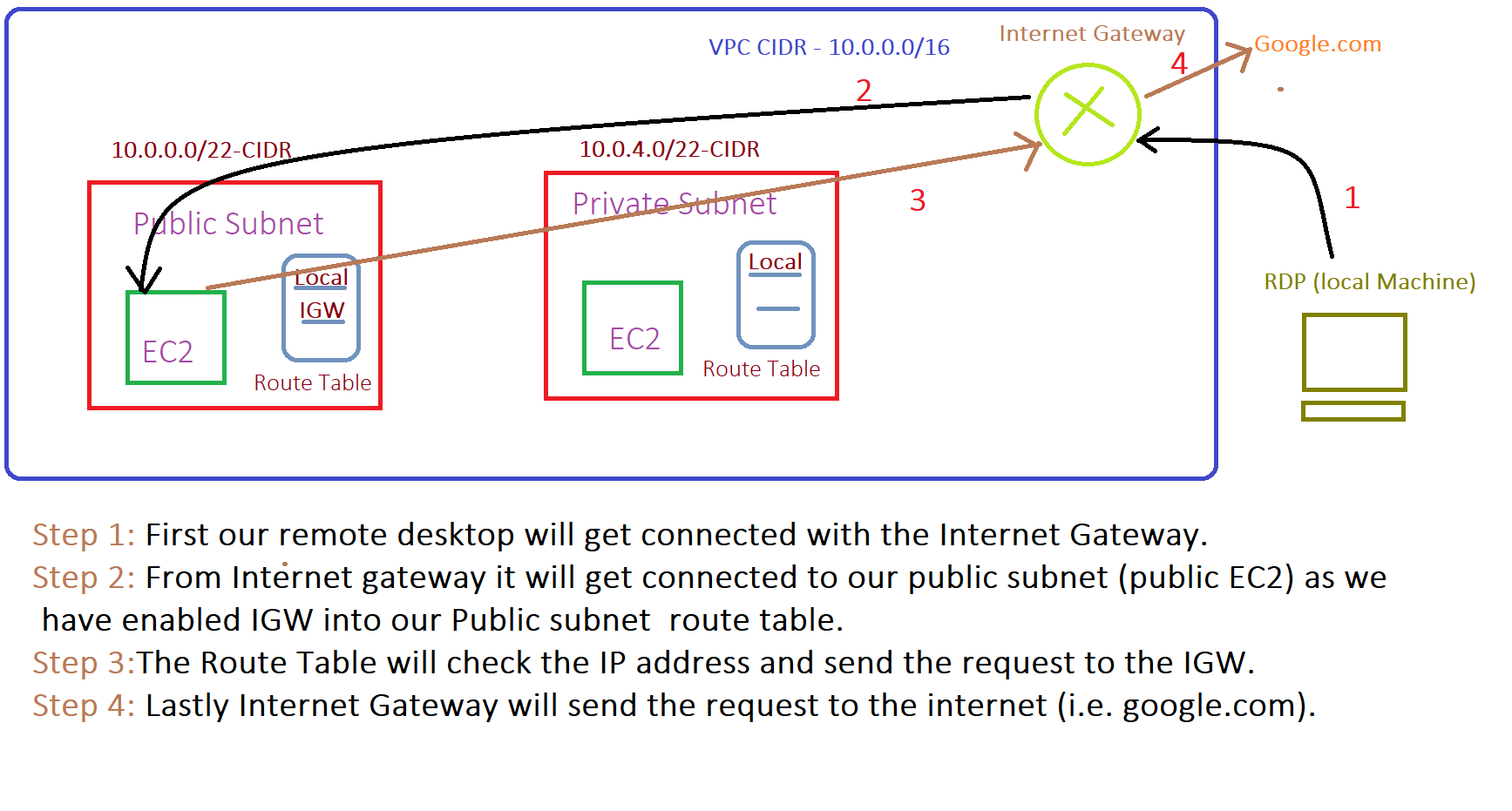




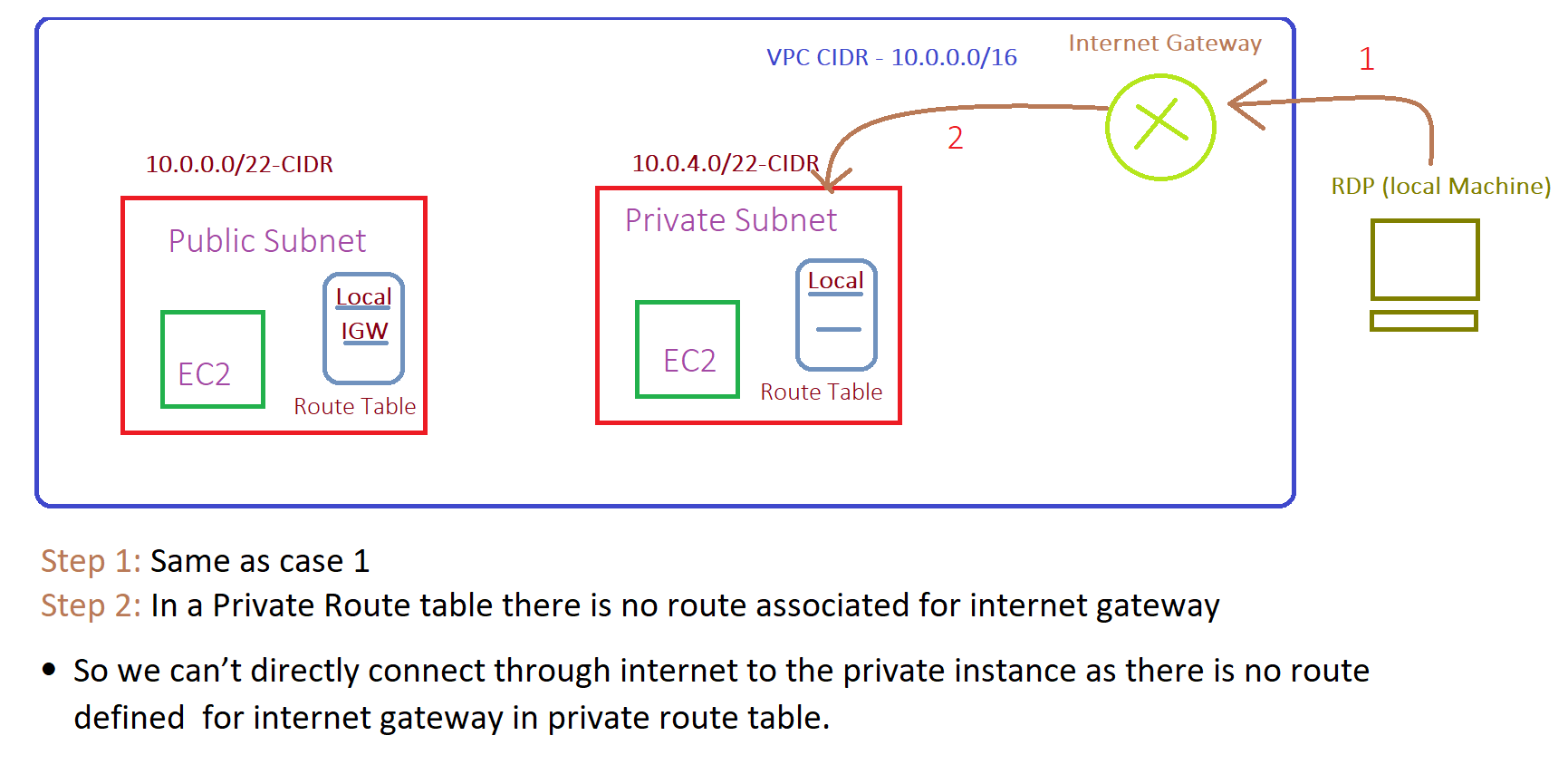


To check if we have the internet connection to our instance or not us this following command into the command prompt of the launched public instance.

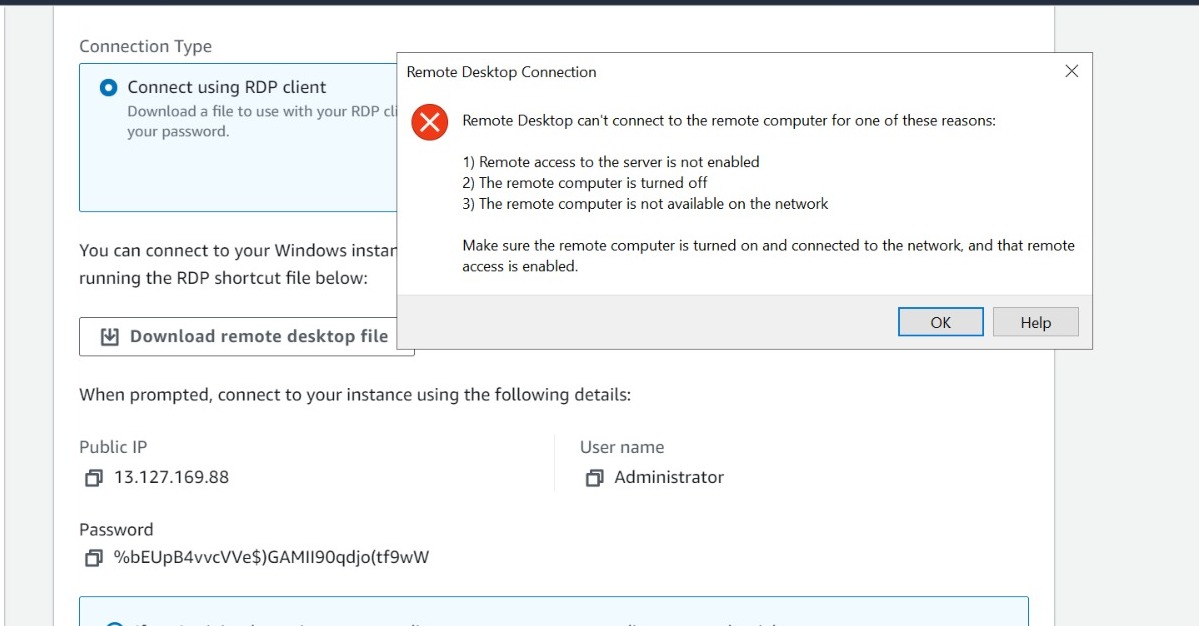
## CASE 1: How we have connected from local Machine to Internet Gateway through (Public Instance).



## Case 2: How we can connect from local machine to Internet Gateway through Private Subnet.



## Now connect to private instance with the same process that we did for public instance

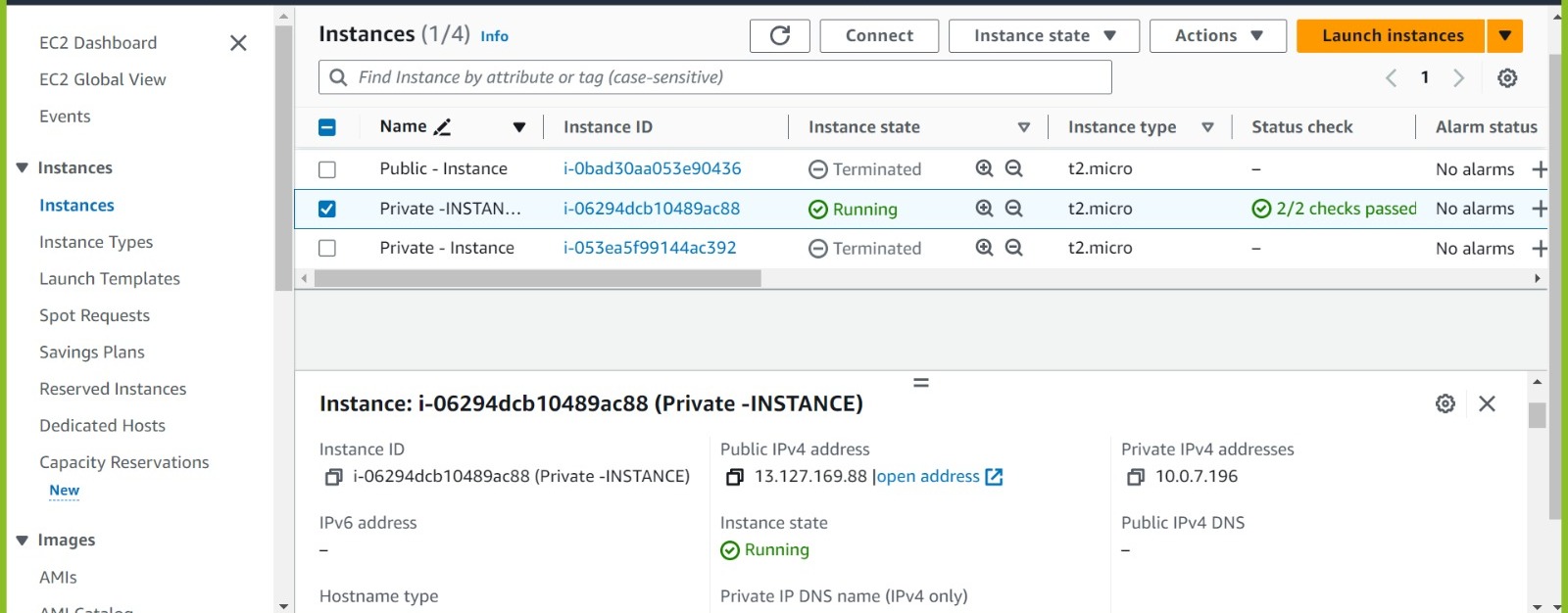


As we can see we are unable to connect to our private instance even though while launching instance we had enabled public IP

***Note :***

* In a Private Route table there is no route associated for internet gateway
* So we can’t directly connect through internet to the private instance as there is no route defined for internet gateway in private route table.

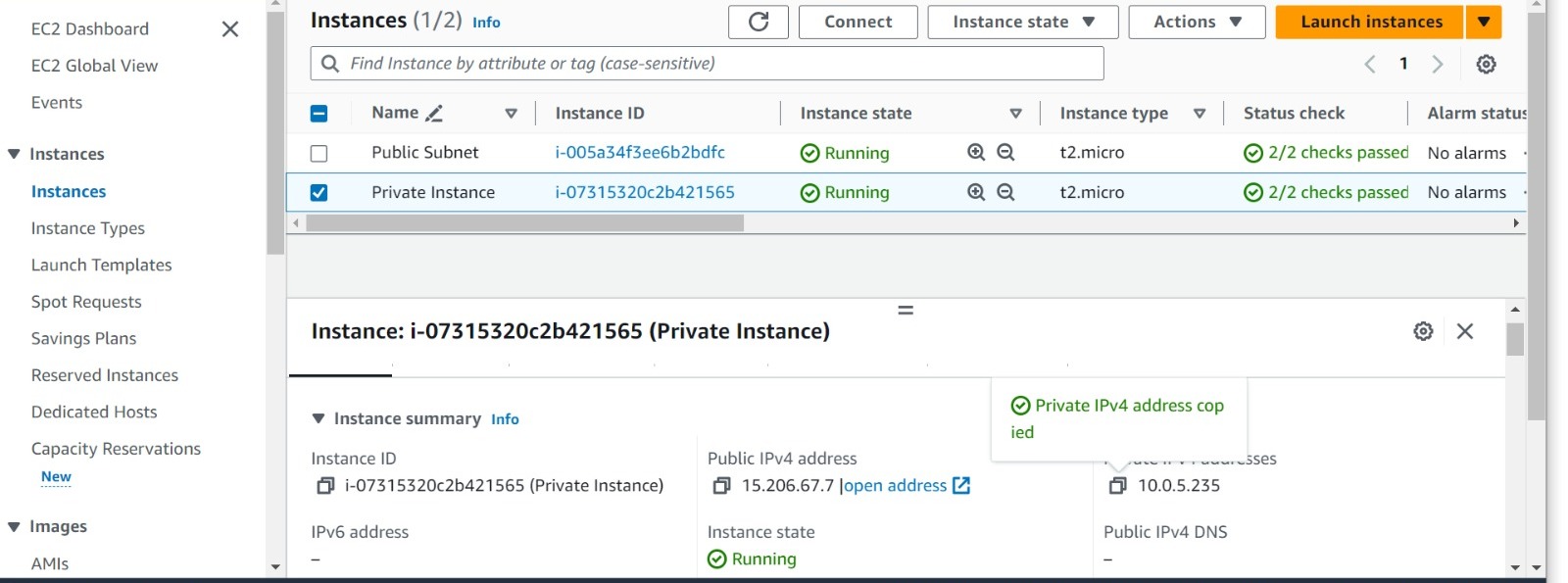
## 



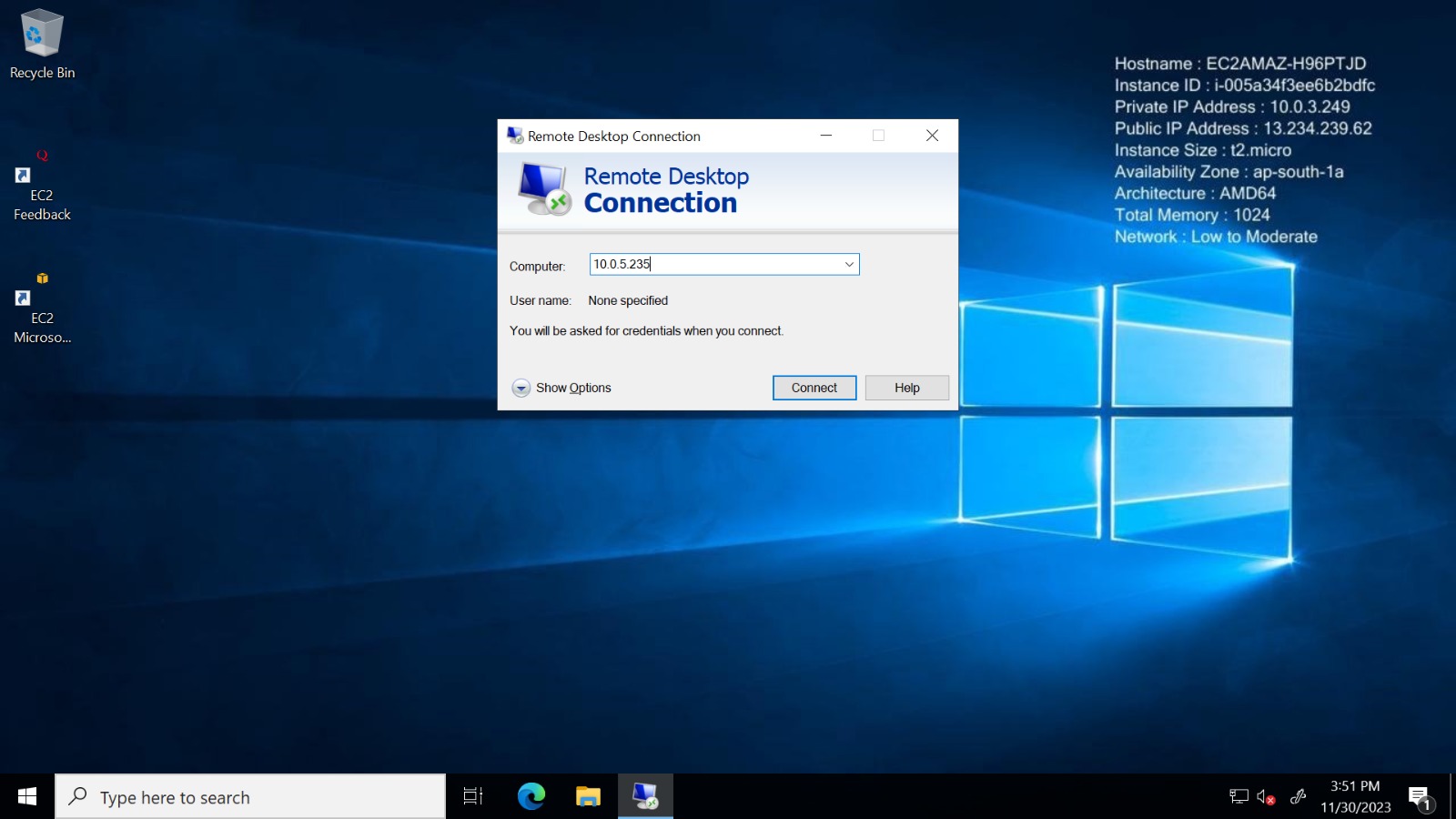
From here we have copied (private instance) public IP and tried to connect from the remote desktop of Public EC2.



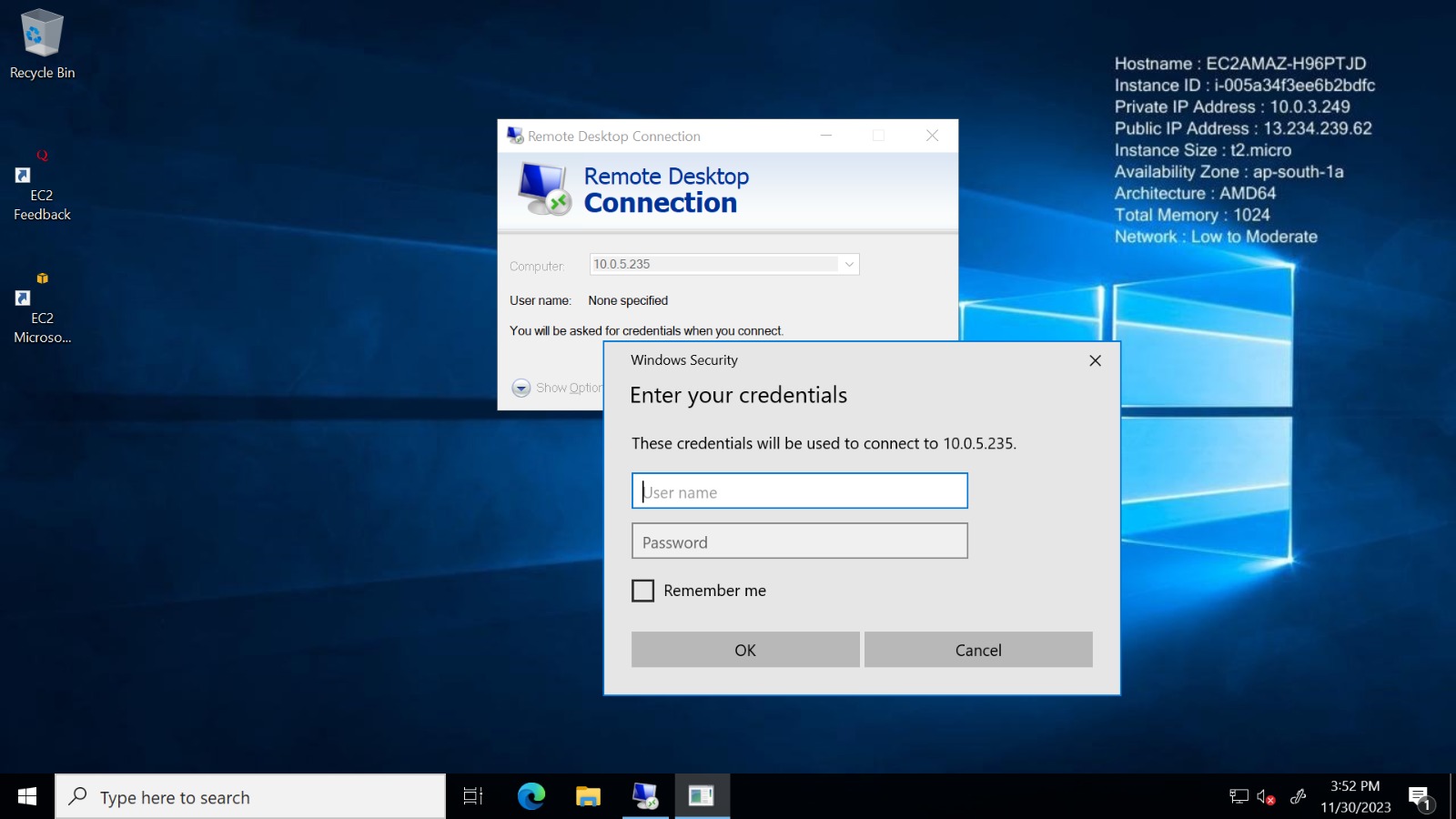
But still we have got the timeout error.



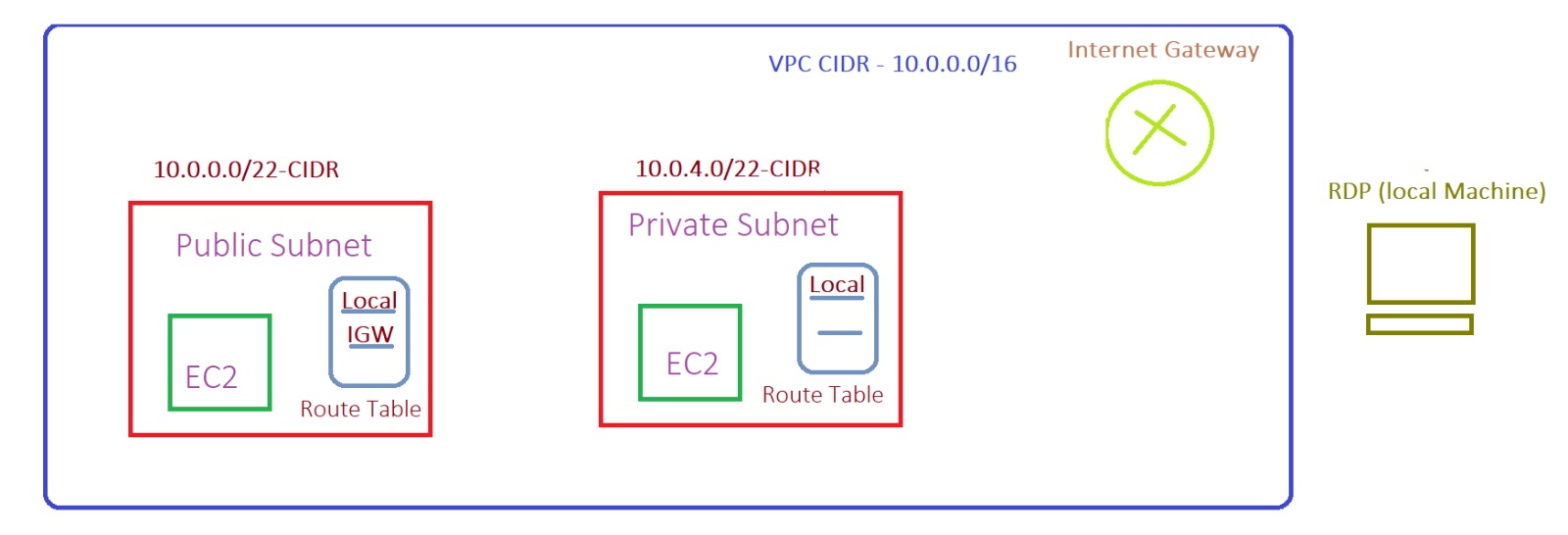
Now Copy the (PRIVATE IP) of Private Instance.



And paste into the RDP of Public Instance

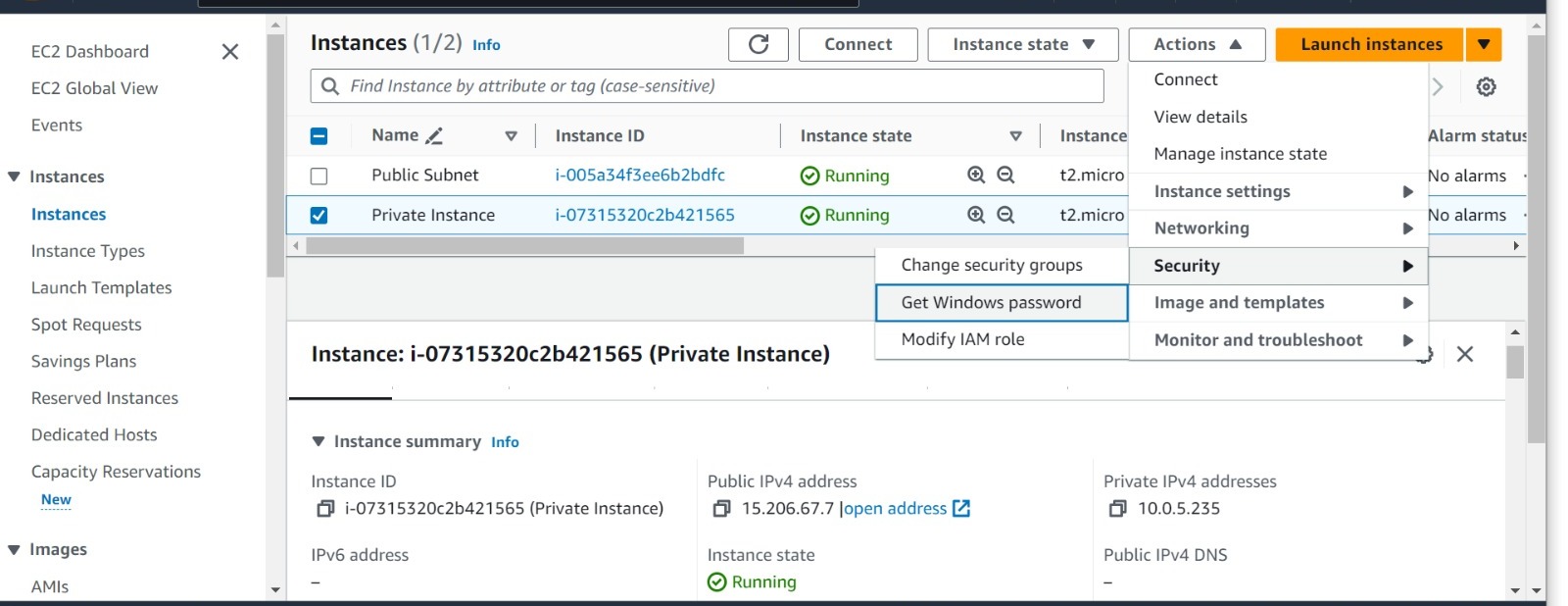


* As we can see we have been able to successfully connect to our private instance
* As local route can connect to any subnet within the VPC.
* Local Route Range – 10.0.0.0/16 –10.255.255.255/16
* Private IP – 10.0.5.235.

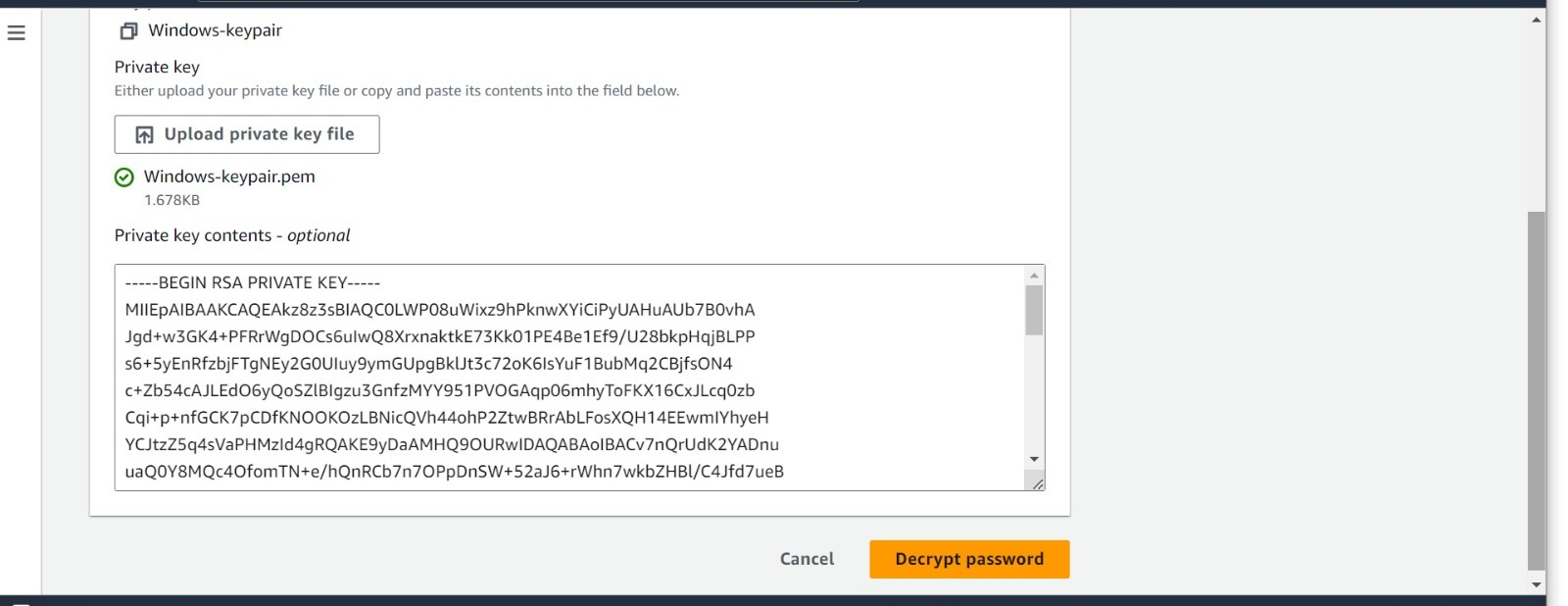




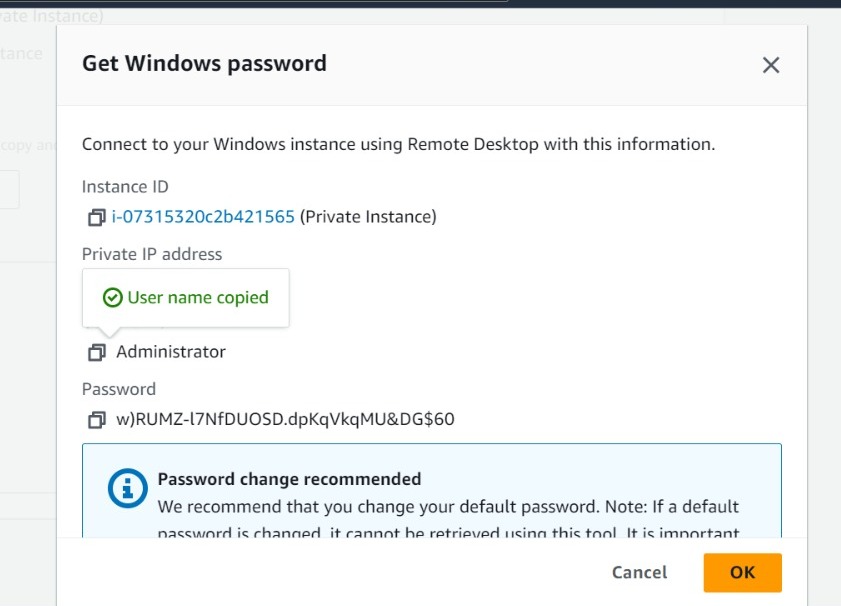
This is because from public instance the (IP of private instance) has followed the route of local route and not the IGW as private IP is in the range of local route .



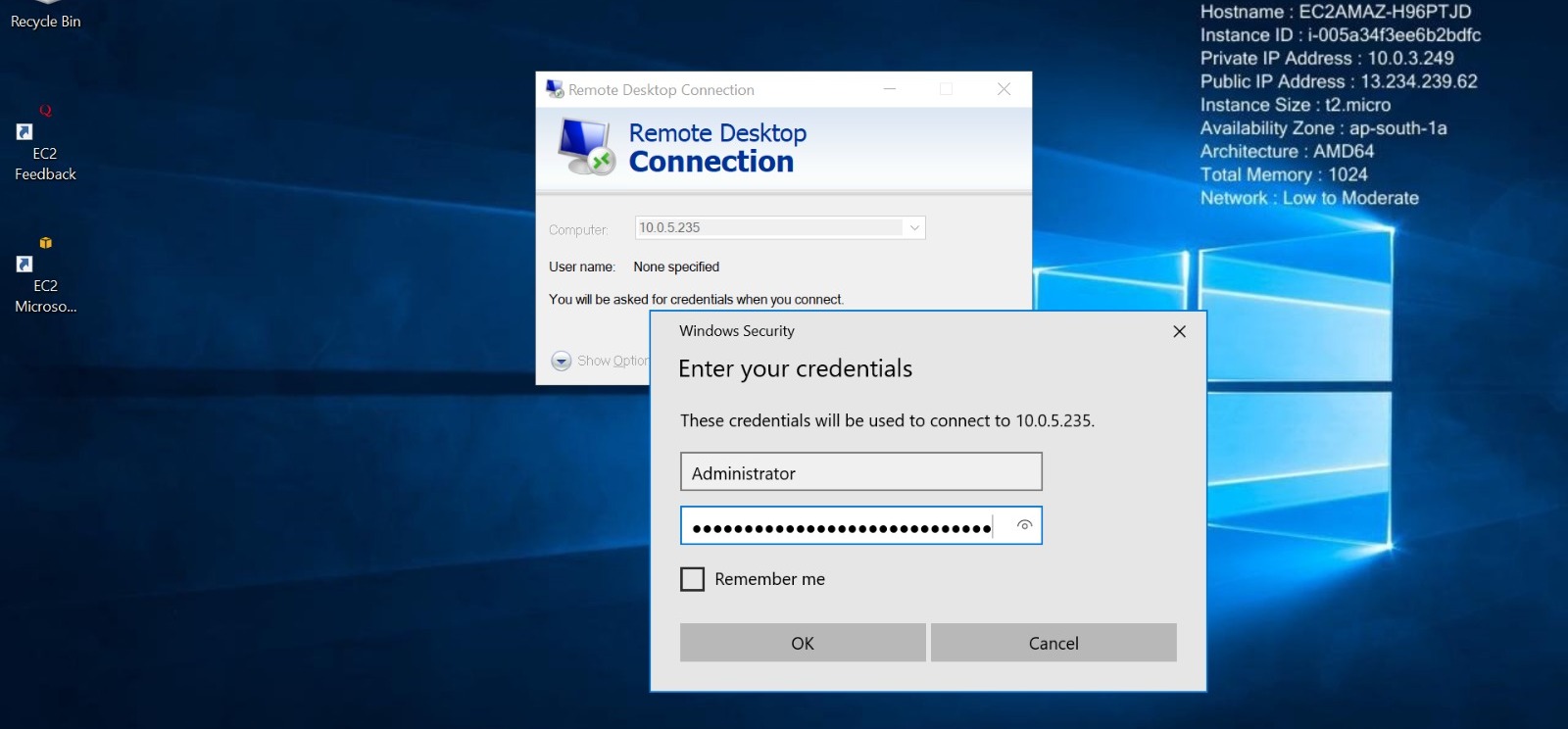
Now get the password for Private Instance to connect through RDP



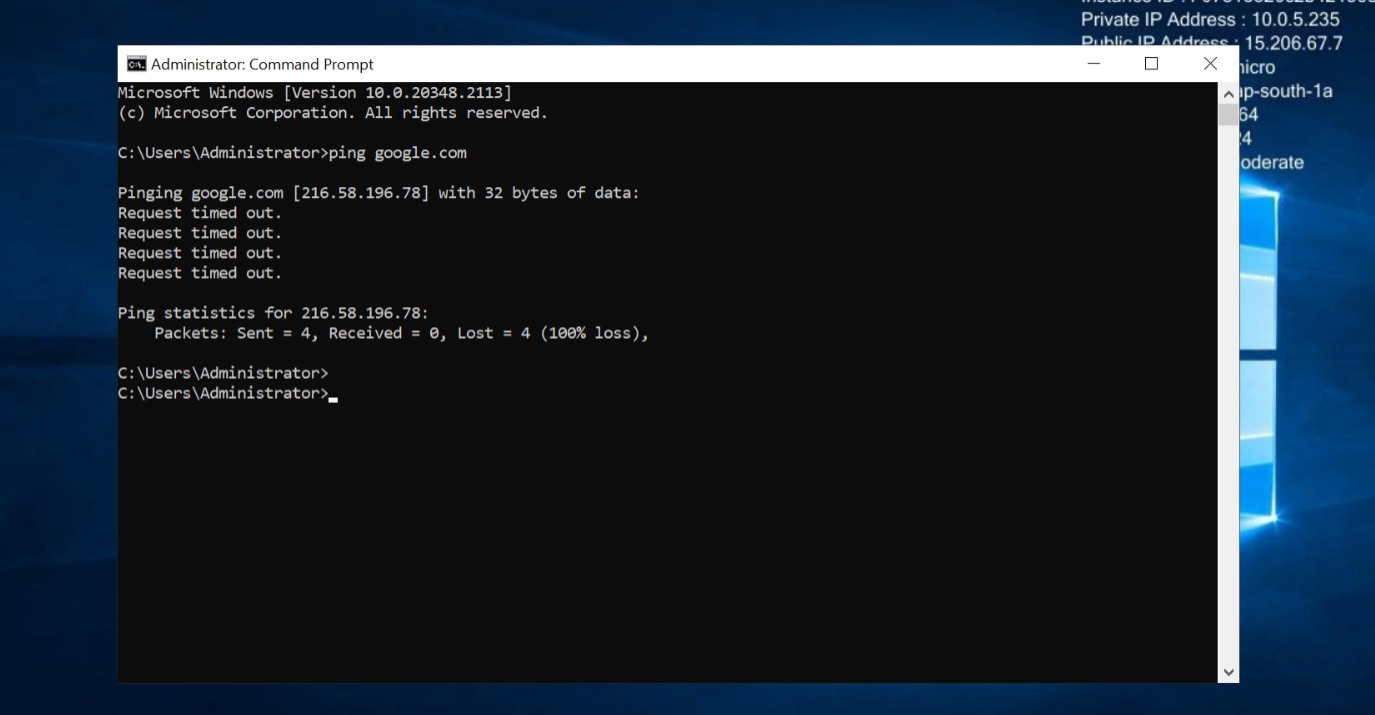
Decrypt it



Copy the decrypted password--> Paste it --> Lastly get connected.



Enter all the credentials over here of Public Instance



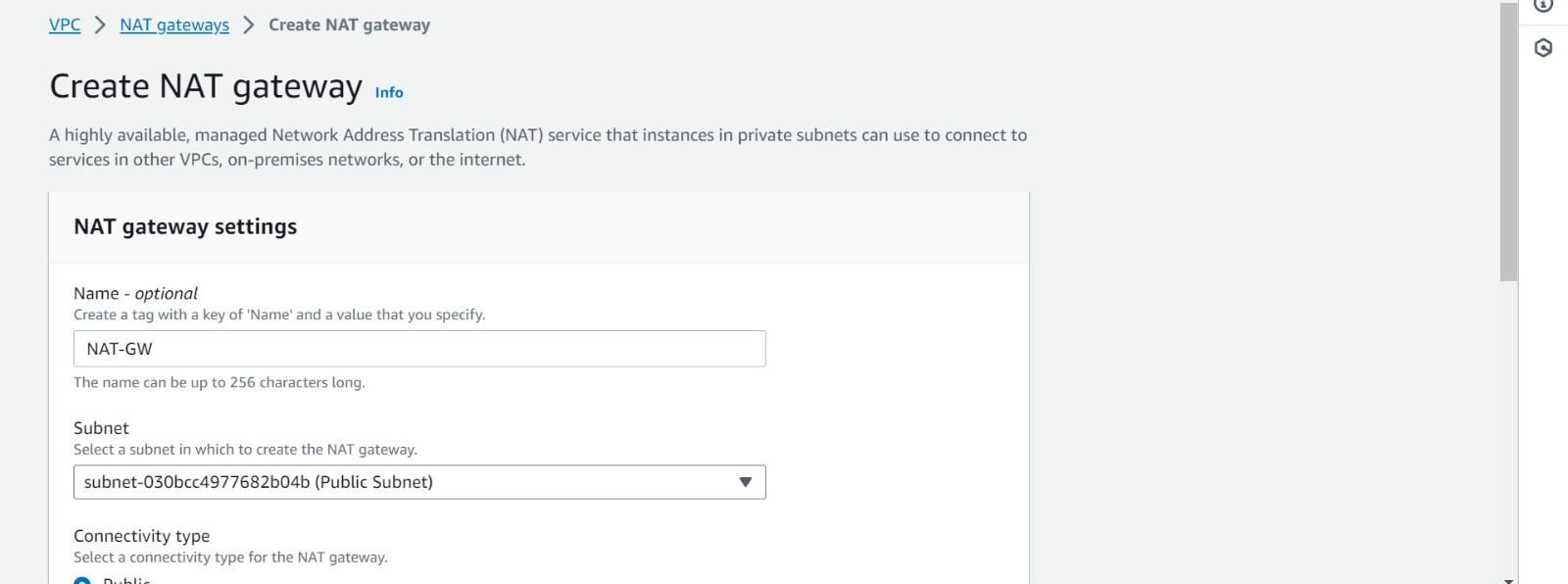


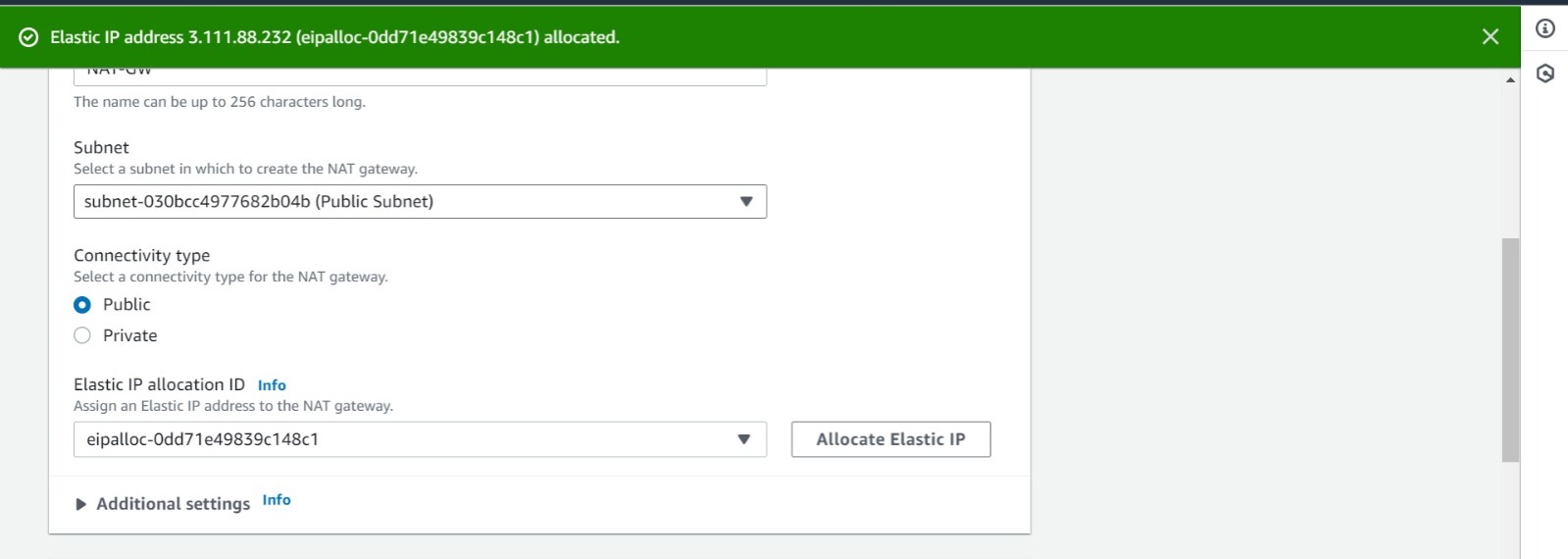
* As still we have connected to our private instance (subnet) successfully but we do not have internet over there.
* Private Instances are mostly used to keep backend databases as we don’t want to lose our data because no-one can access our private instance as we have not provided internet to this instance .
* As you can recover application but not the data.
* You cannot connect to private instance via private

## NAT Gateway

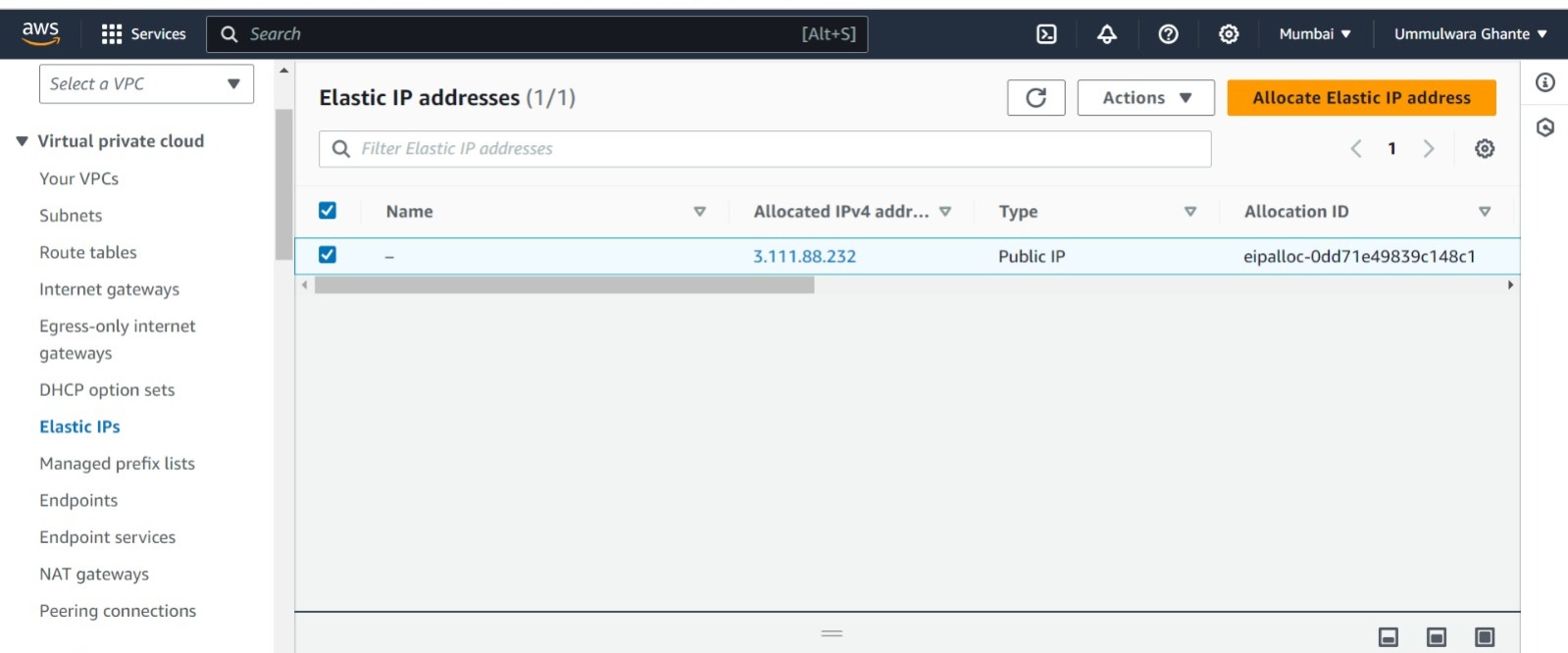
* NAT Gateway is used to provide internet to private subnet through public subnet.
* NAT GW is associated with the public subnet because we cannot provide direct internet to the private subnet so NAT act as a mediator between internet gateway and private subnet.
* Private subnet sends the request to NAT gateway through private (NAT Route) then this further connect to the public subnet route IGW and the IGW connect with the internet in this way we will get the internet connection to private instance.
* Here NAT will only communicate with internet and respond back to private subnet through internal IP (i.e., private IP) that is not known by the internet.
* NAT has been allocated with an elastic IP so that it can communicate to the internet and internet will respond back to the same IP (i.e., elastic IP) so that internal IP of private subnet is hidden inside the NAT gateway and it is not visible to the internet (outside world).
* Whatever the request asked by the private subnet to NAT GW is sent back to it through internal ip of private instance (i.e., private IP).
* Elastic IP act as a public IP that’s why NAT gateway can communicate to the internet.
* Public/elastic IP is only visible to the outside world not the hidden IP (i.e. private IP).

## Now Connect NAT Gateway to public subnet and allocate elastic IP with it.

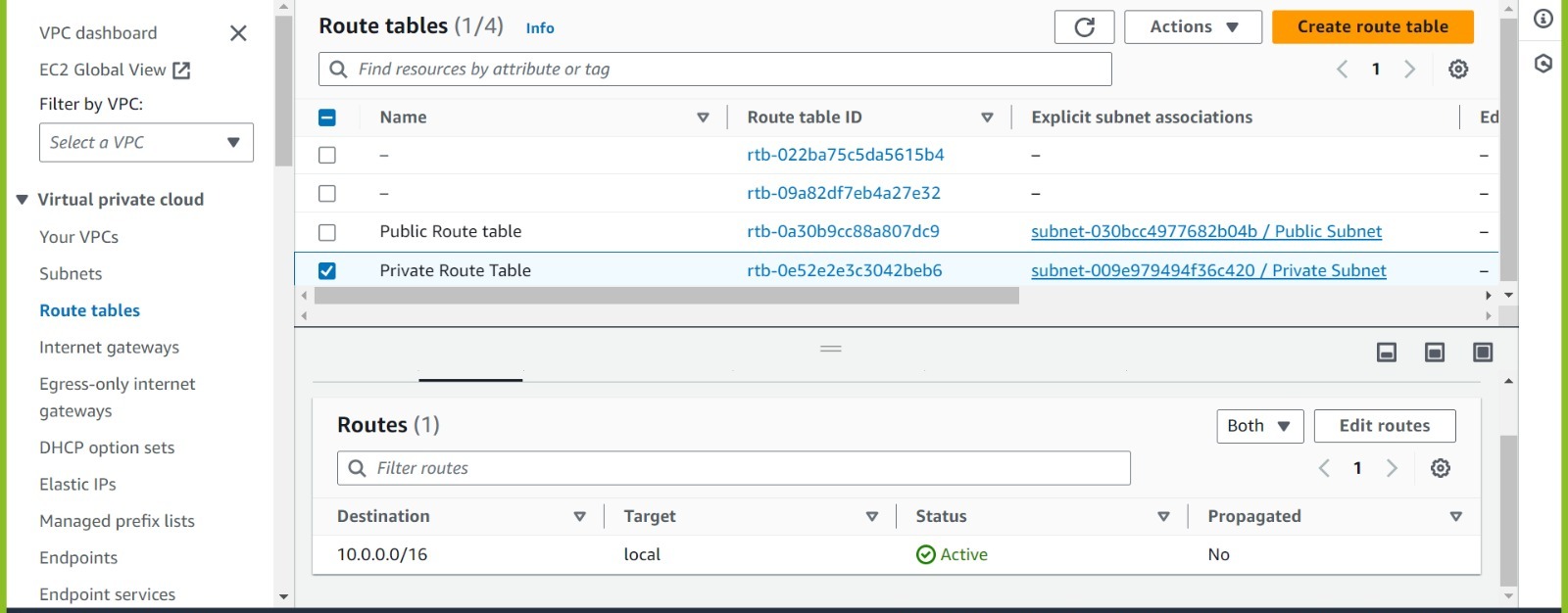


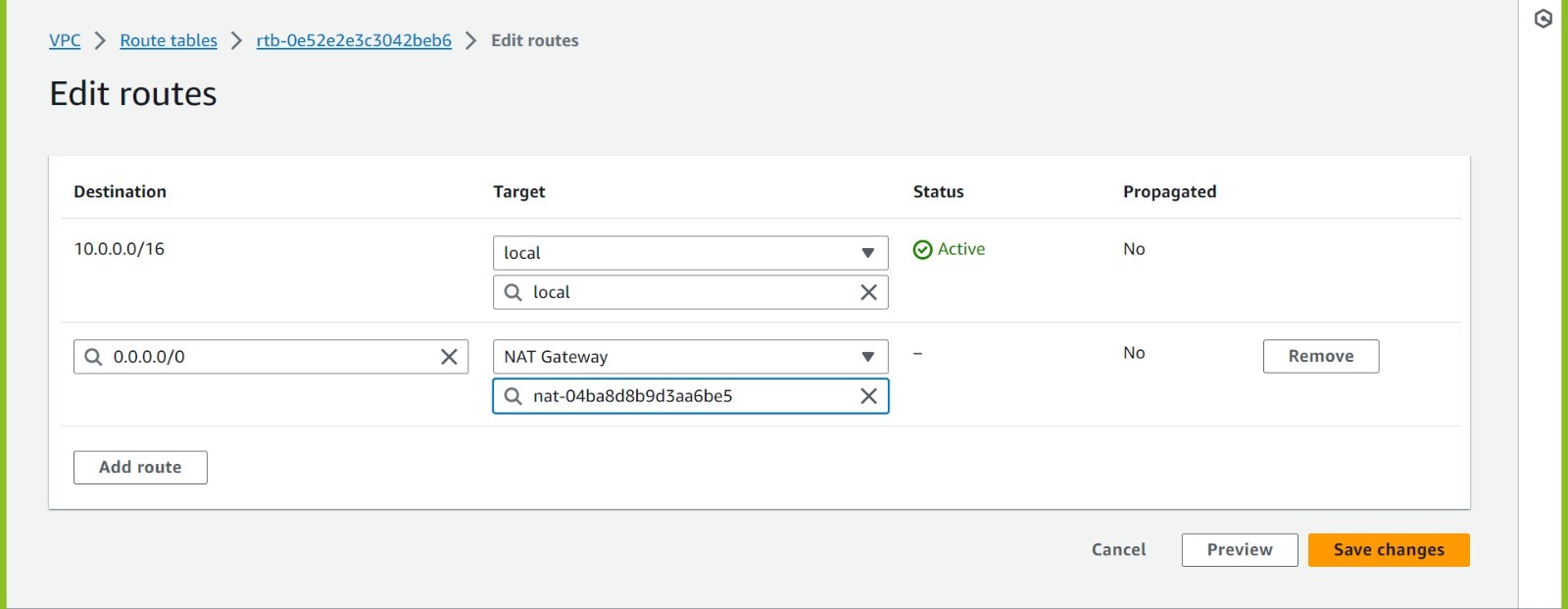


From here we can generate and allocate elastic IP to our NAT gateway.

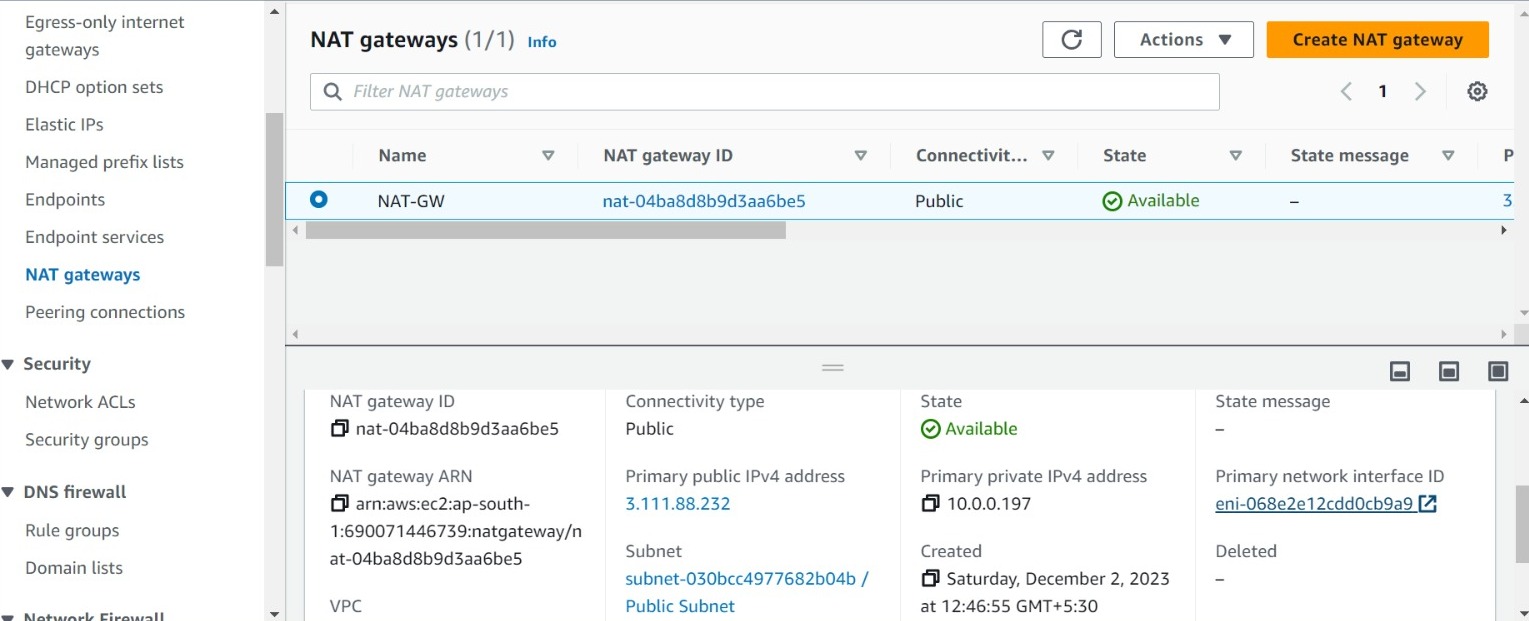


Elastic IP has been generated successfully.

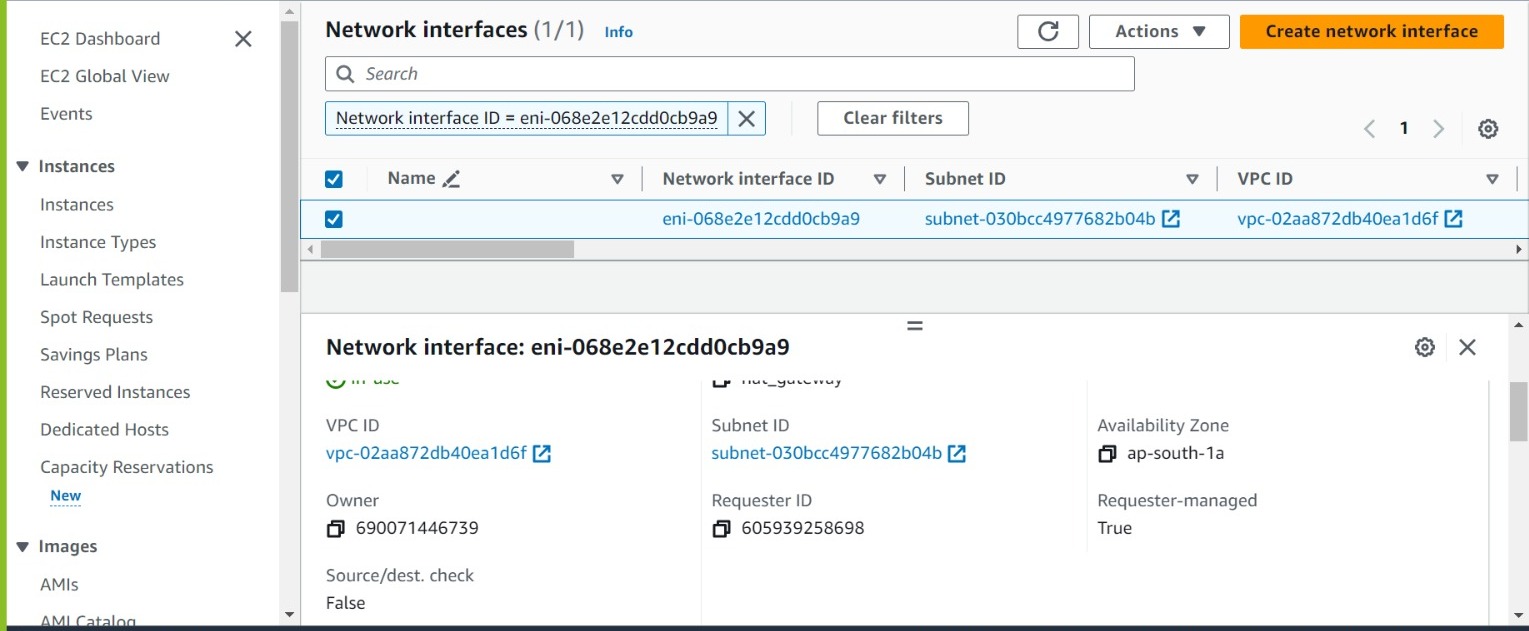




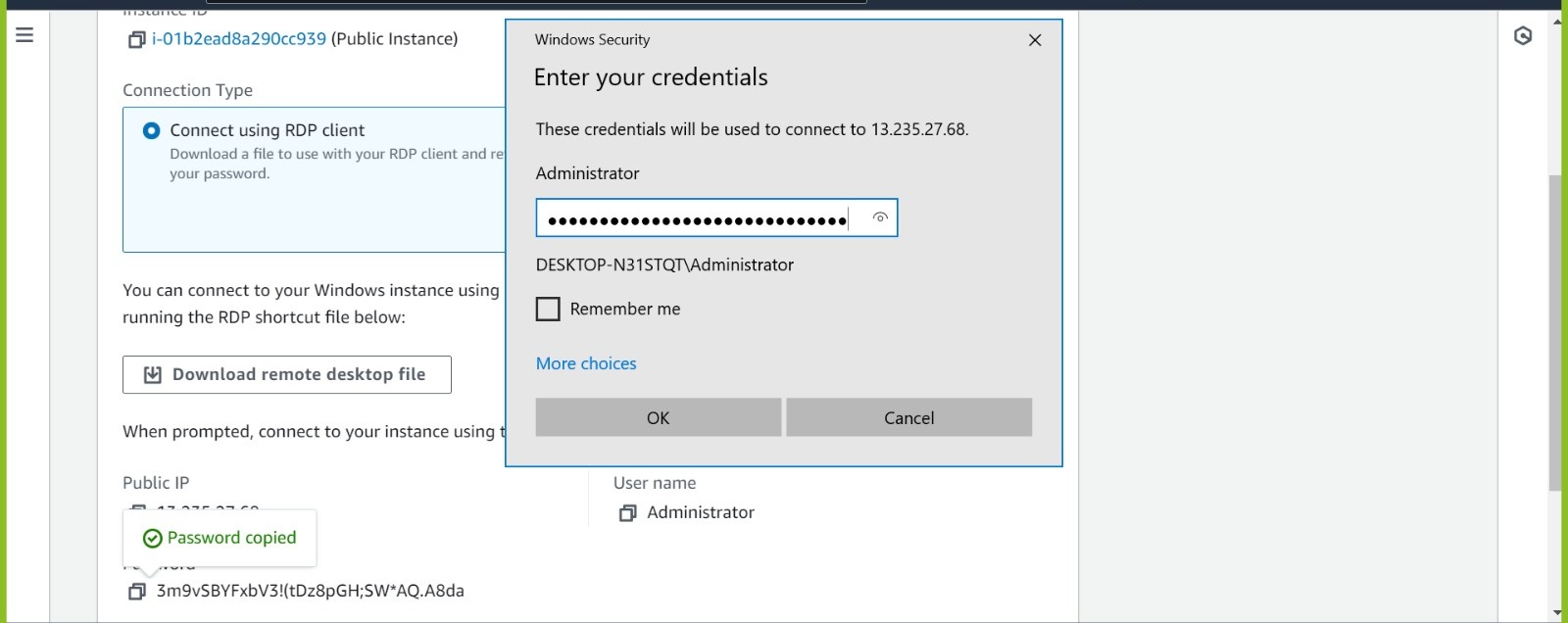
Here in private route we are giving permission to give access to internet through NAT GW route so NAT will connect to the IGW route of public subnet and through IGW we will get the internet.



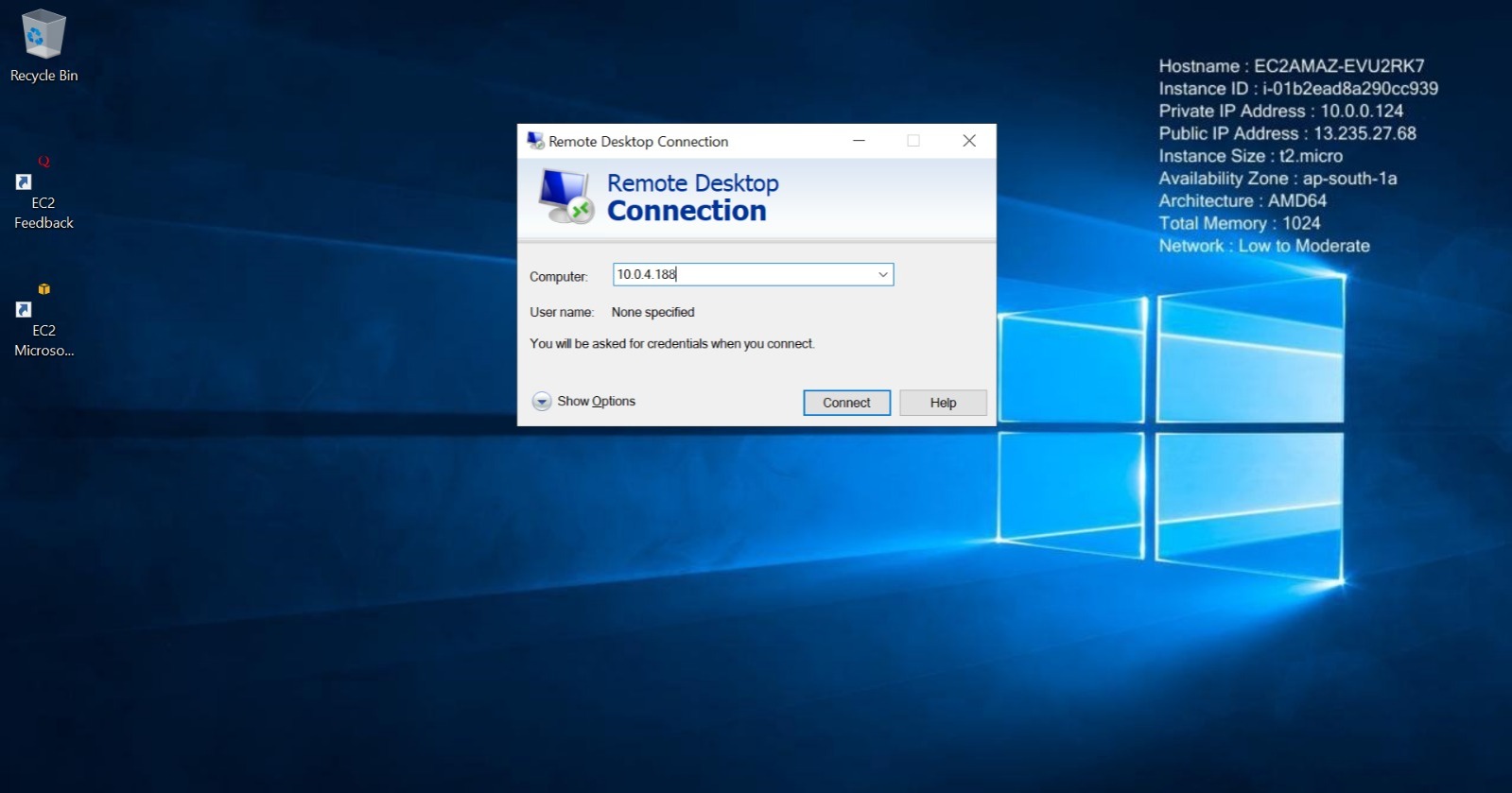
As we can see this is the NIC card attached to our NAT GW.



As we can see here following is our public subnet ID.

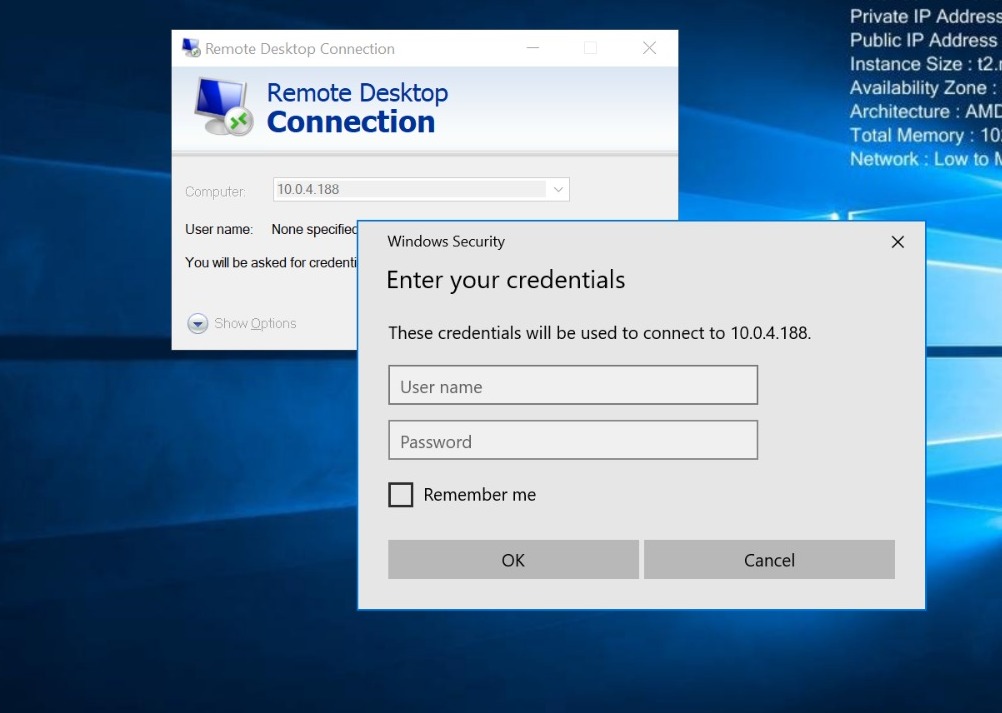


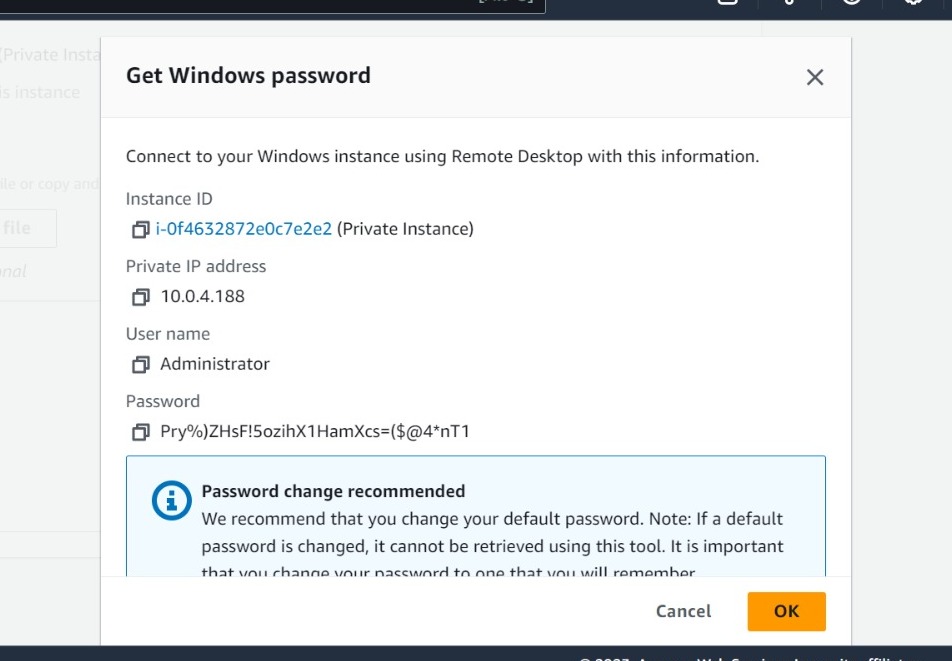
Now again connect to the public subnet through RDP.



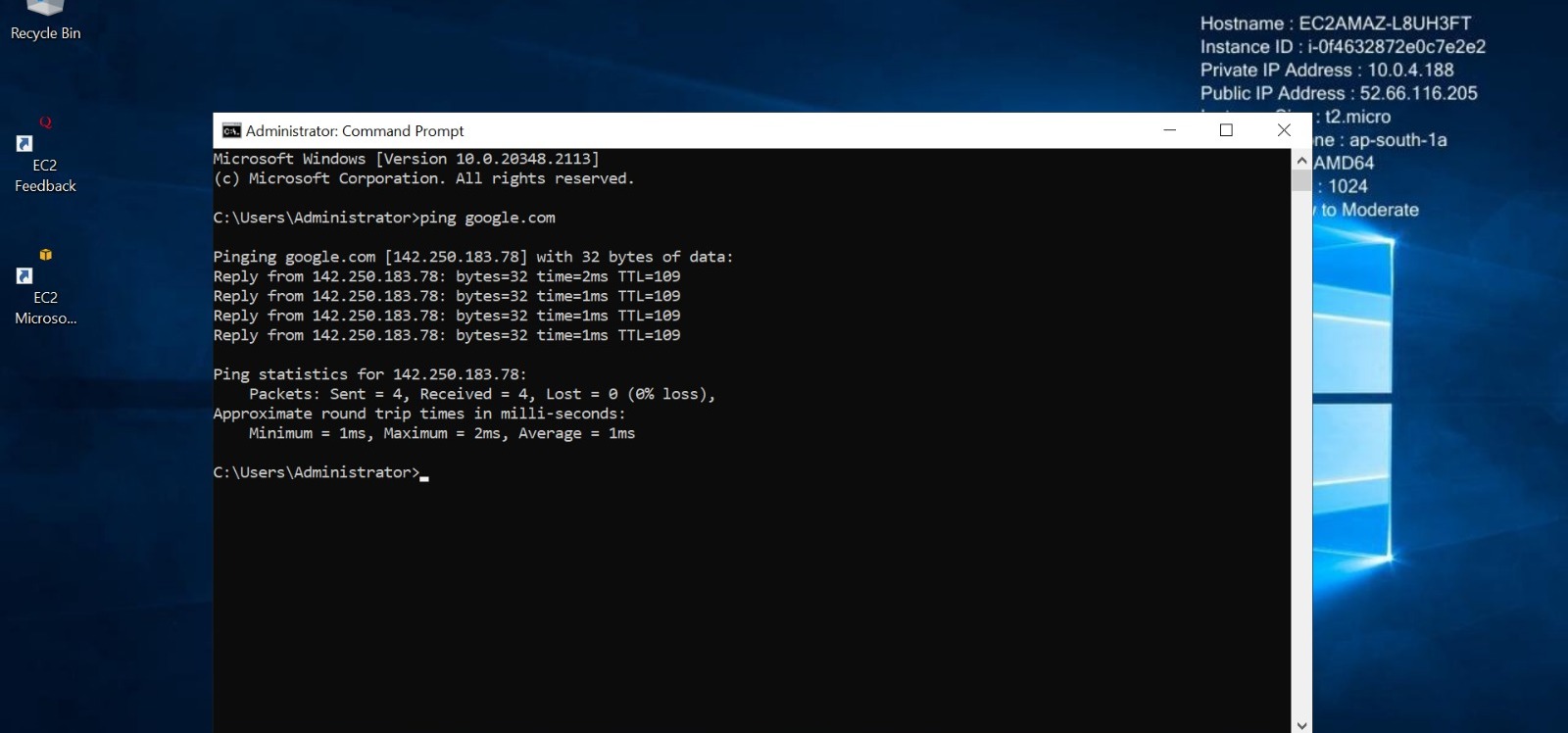
Successfully connected to the public subnet.

In public subnet through RDP, we can connect to private subnet by just pasting private IP.

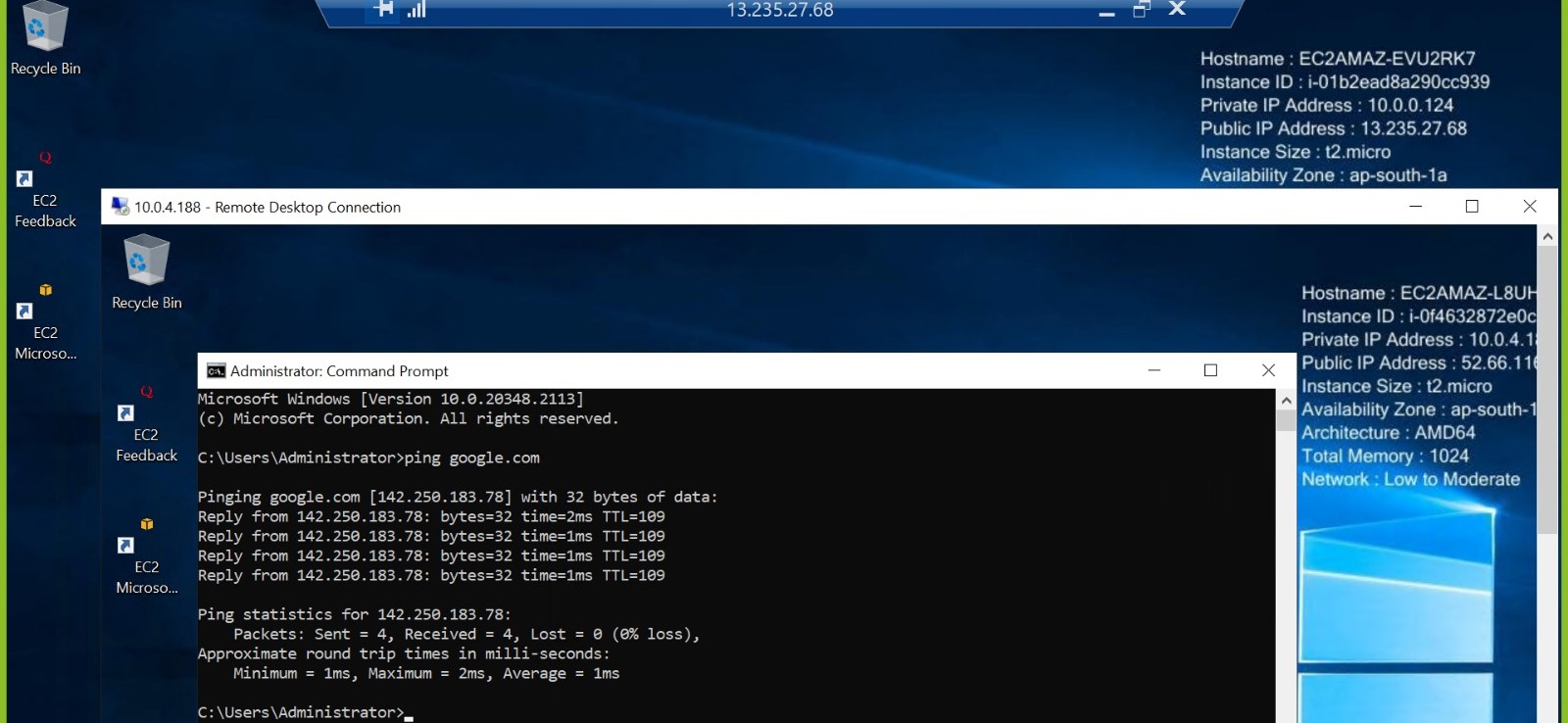




Now enter all the credentials and get connected to the private subnet.



* As you can see we have successfully connected to the private instance and we have successfully connected with the internet.
* So by concluding this we can only connect to the internet the outside world cant connect to us as this is only one way communication as private subnet can connect to the internet but internet itself cannot connect with the private subnet.
* As we have connected from public instance to private instance so the public instance is known as JUMP SERVER or also called BASTION HOST.



From here you can see it.