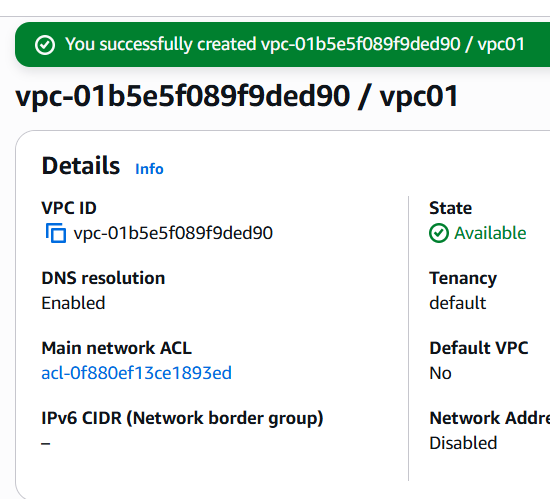
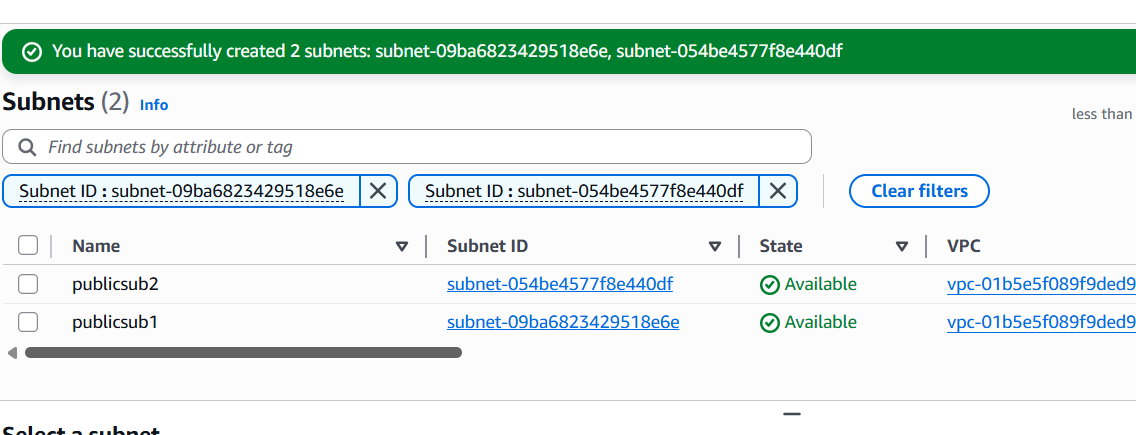
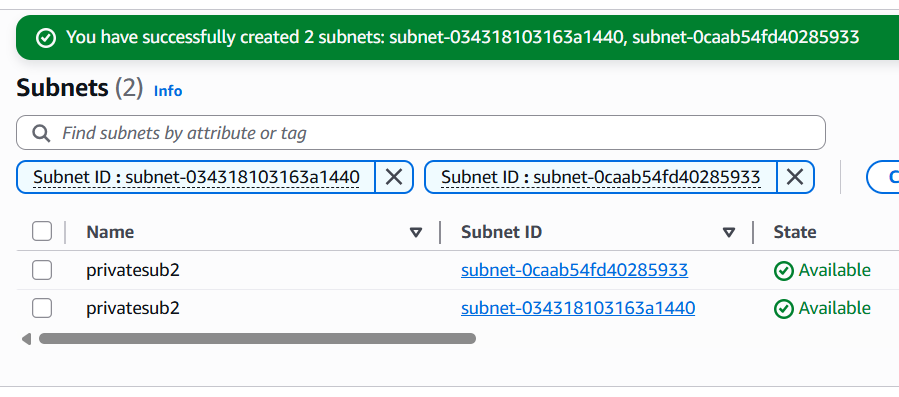
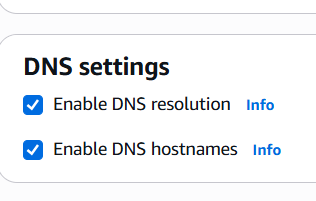
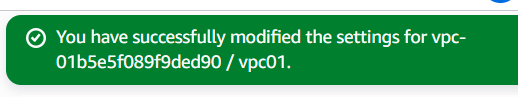
**VPC**

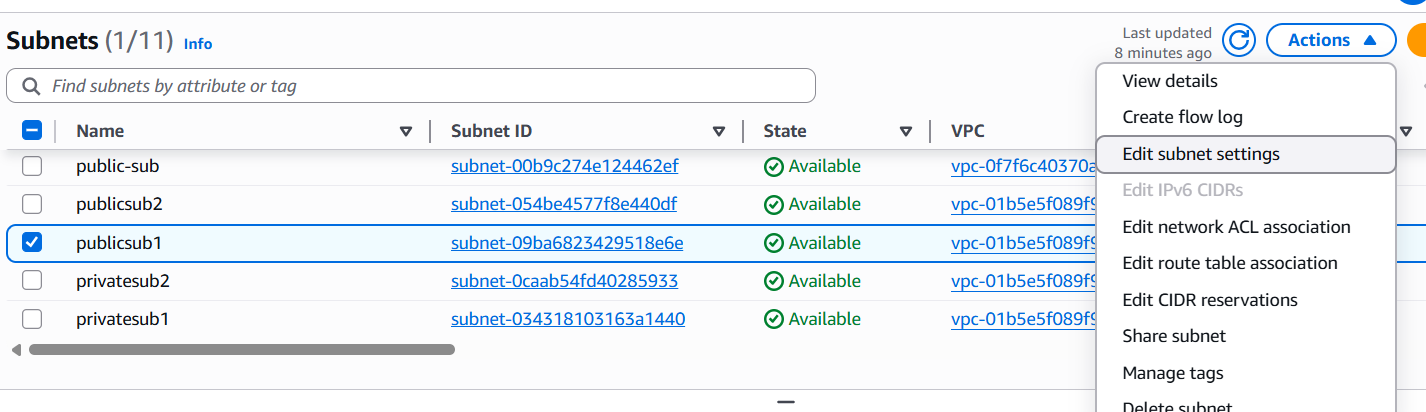
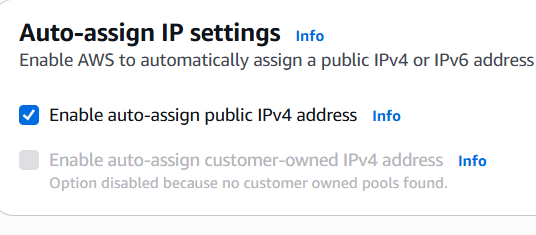
1 . Create VPC with 2 private and 2 public subnets.

* Login to aws console
* Search for VPC
* Click create vpc
* Select vpc
* Give CIDR range
* 
* Now select subnets
* Create subnets
* Subnet name , availability zone select us east a1
* Give ipv4 subnet cidr block
* As 11.0.0.0/28
* 2 public subnets
* Publicasub1-10.0.0./24
* Publicsub2-10.0.1.0/24
* 
* 2 private subnets
* Privtesub1- 10.0.2.0/24
* Privatesub2-10.0.3.0/24
* 

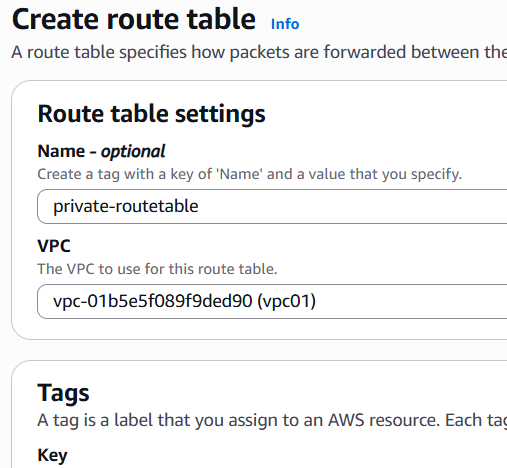
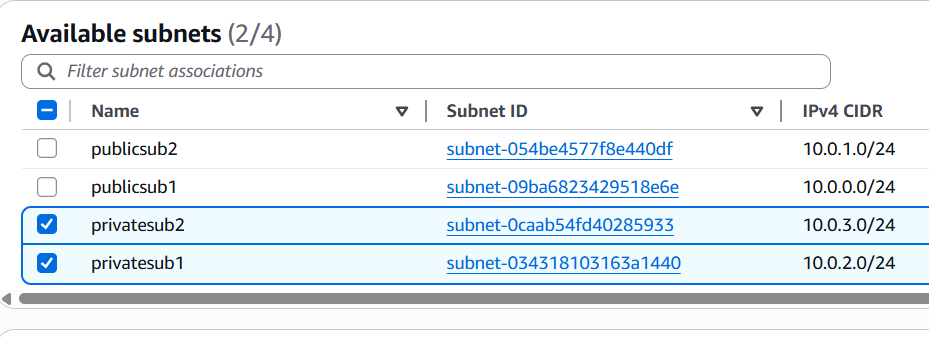
2. Enable DNS Hostname in VPC.

* Go to vpc page
* My vpc
* Select vpc and click actions hit edit vpc settings
* Enable dns resolution and enable dns hostnames in dns settings
* 
* 

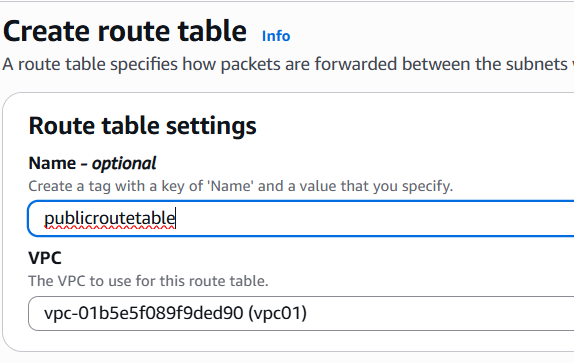
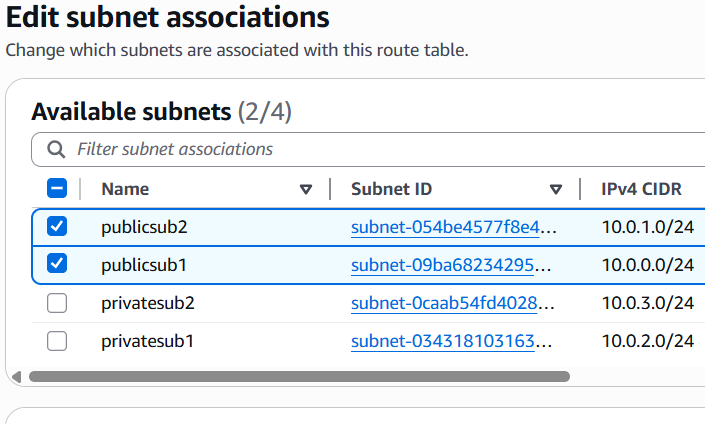
3 . Enable Auto Assign Public IP in 2 public subnets.

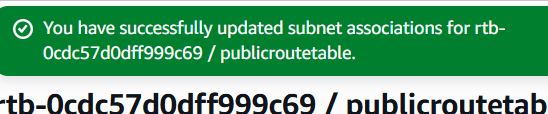
* Clicks subnets in vpc
* Select the publicsub1 and go to actions and click edit subnets settings
* 
* Enable auto assign ip
* 
* Publicsubnet2

4. Add 2 private subnets in private route table.

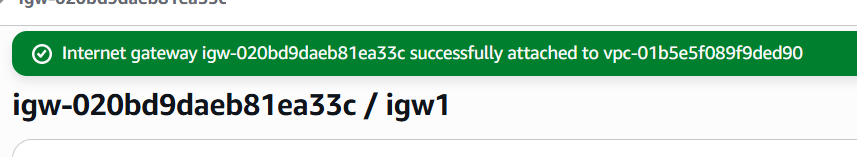
* Go to vpc dashboard
* Navigate to routetable
* Create route table
* Add your vpc
* 
* Adding 2 private subnets in private routetable
* 

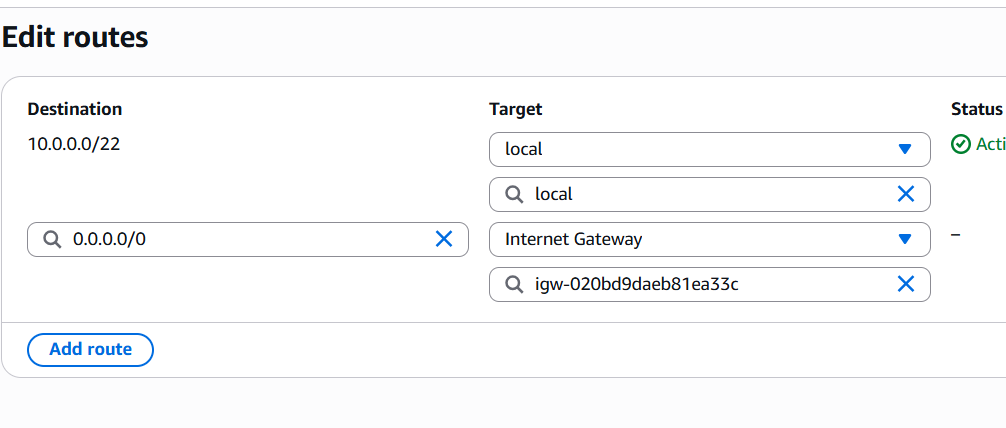
5. Add 2 public subnets in public route table.

* Creating one public rfoute table
* 
* Adding public subnets in public route table
* 

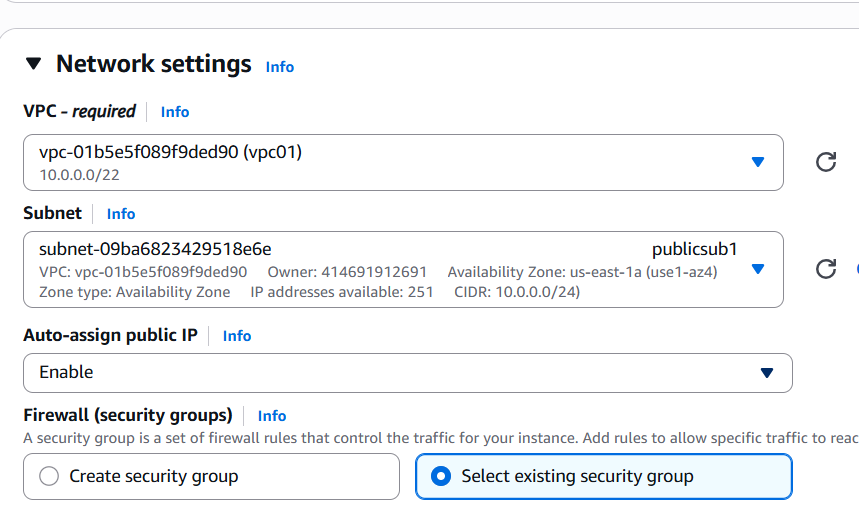
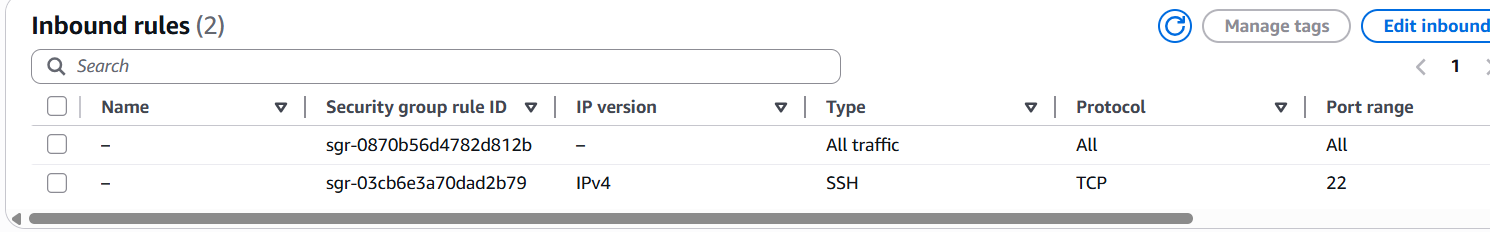
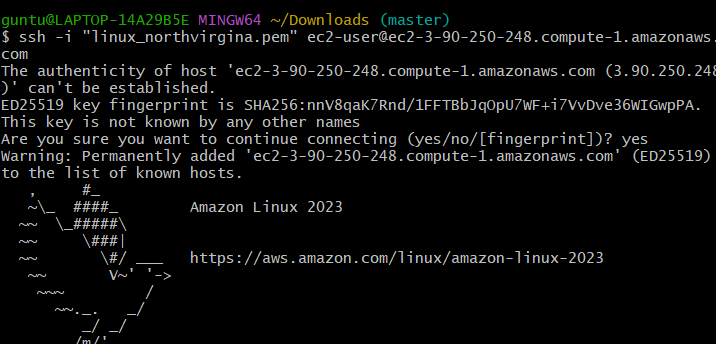
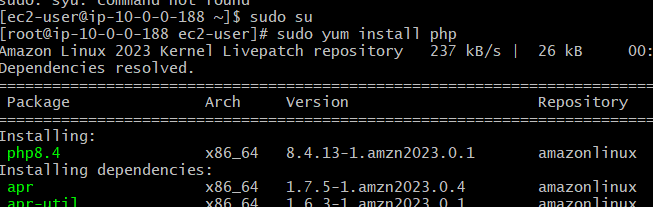
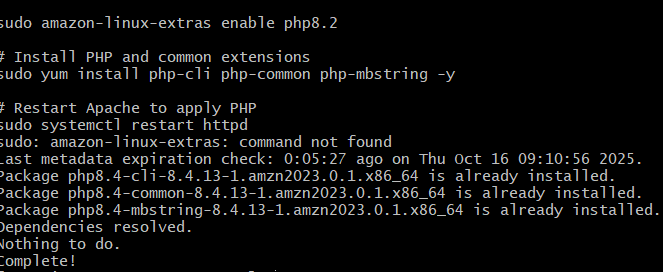
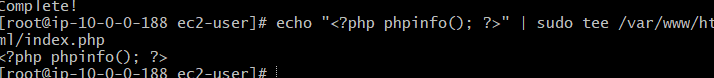
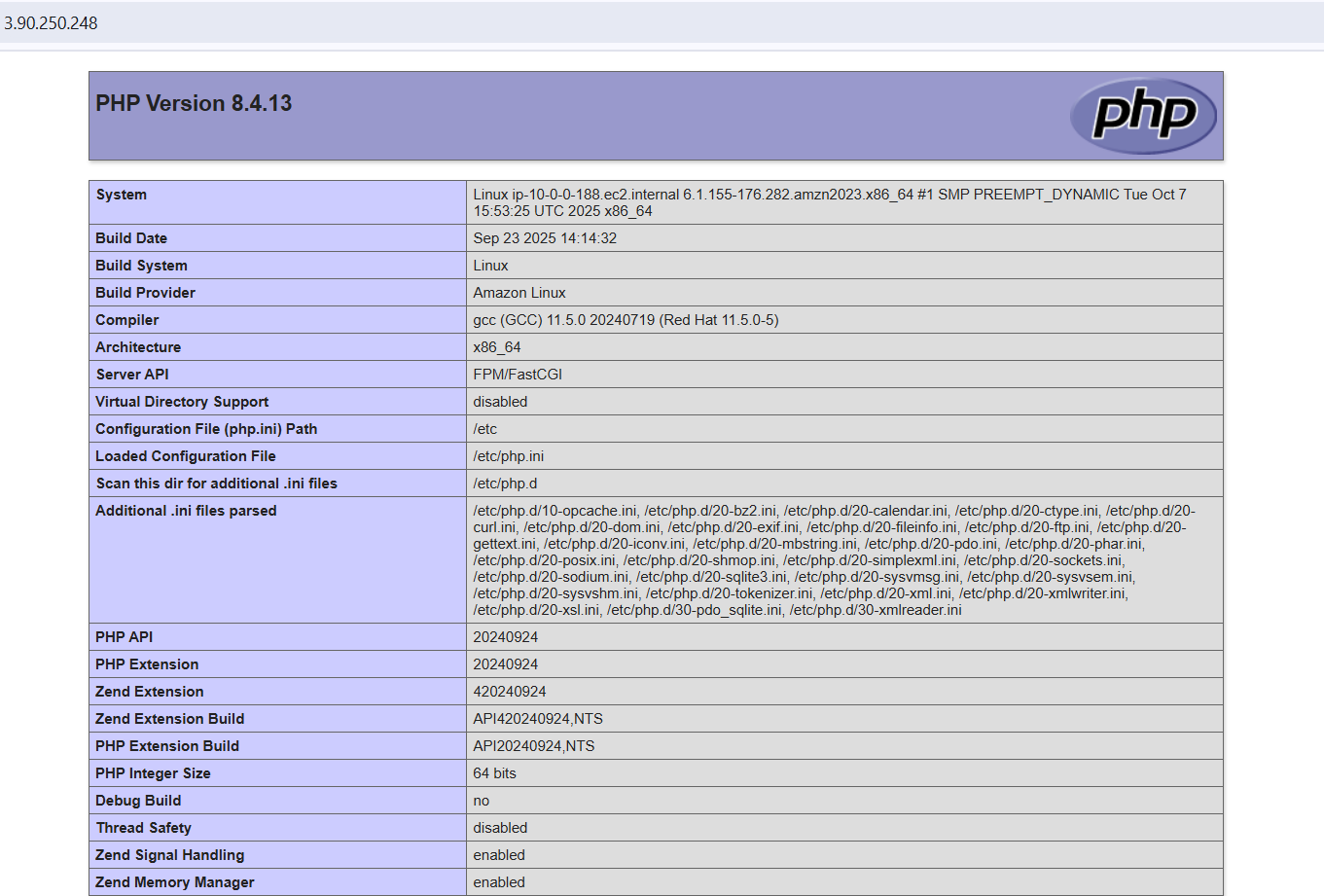


6. Public route table will have the routes to internet and local.

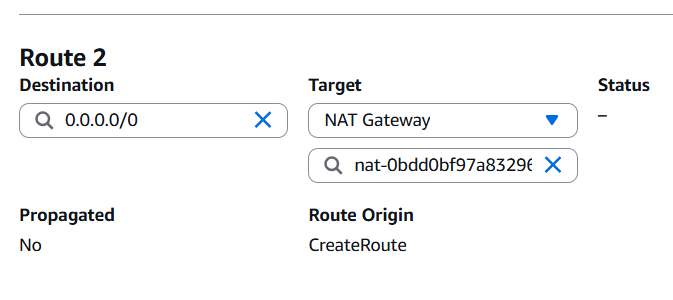
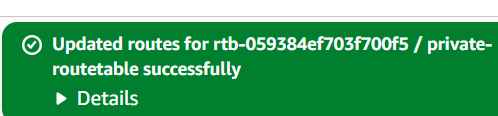
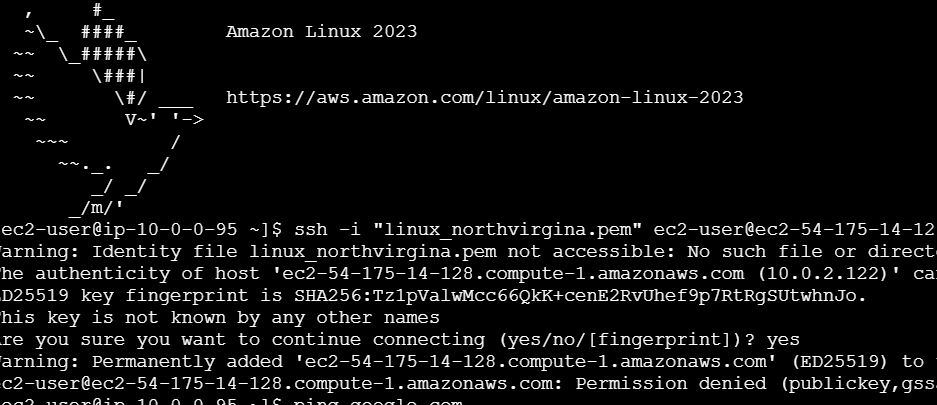
* Go to vpc dash board
* Select internet gateways
* Create internet gateway
* Now attach this internet gateway to VPC
* 
* Now, navigate to route table
* Select the table
* Now edit route table
* Add route
* Allow 0.0.0.0/0 and select created internetway
* \



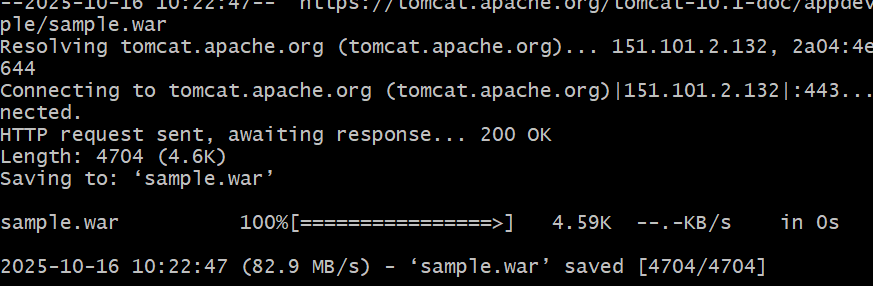
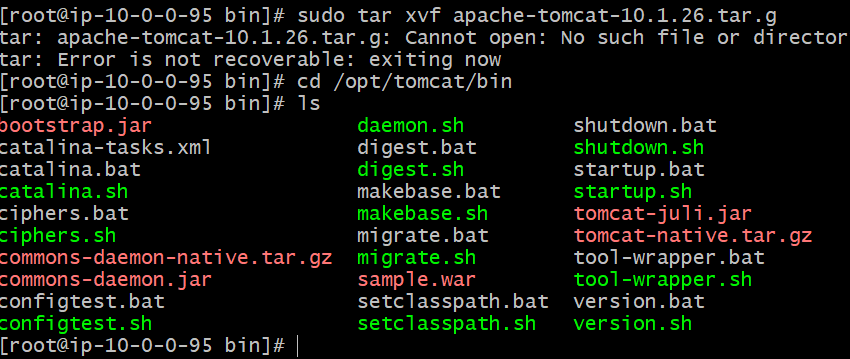
7. Create EC2 in public subnet with t2.micro and install PHP.

* Go to aws console
* Open ec2 service
* Launch an instance with created vpc
* Add public subnet
* Enable public ip
* Default security groups
* 
* Launch an instance
* Add port ssh 22 in security groups , inbound rules
* 
* Connect to instance using pemkey
* 
* Install php
* 
* For php we need to install httpd because, httpd is webserver,
* To see in web page , httpd is needed
* Php alone cant be seen
* Now
* # Enable PHP 8.2 (you can choose another version)
* sudo amazon-linux-extras enable php8.2
* # Install PHP and common extensions
* sudo yum install php-cli php-common php-mbstring -y
* # Restart Apache to apply PHP
* sudo systemctl restart httpd
* 
* Create a test page
* echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/index.php
* 
* Allow port 80 in bound rules
* Now verify in your browser
* <http://(instance> public ip)
* 

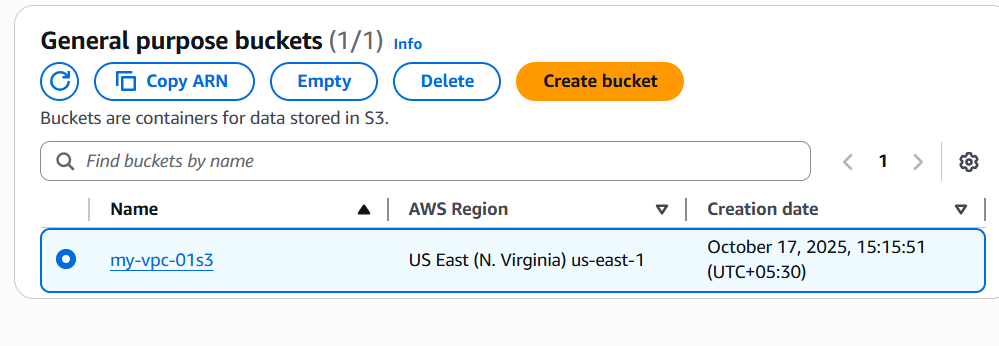
8. Configure NAT gateway in public subnet and connect to private instance

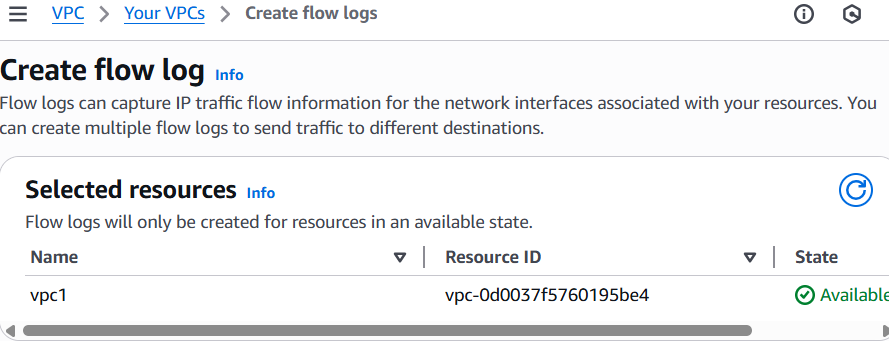
* Create a nat gateway
* Select public subnet
* Allocate elastic id
* Now go to route table
* Edit route table and add
* Add created nat gateway
* 
* 
* Run public subnet instance
* Run private subnet instance
* Using pem.key connect to publicsubnet instasnce
* Now after connection has been established
* Enter private subnet instance pem.key in public.subnet instance
* 

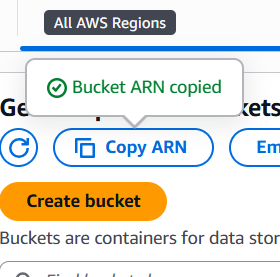
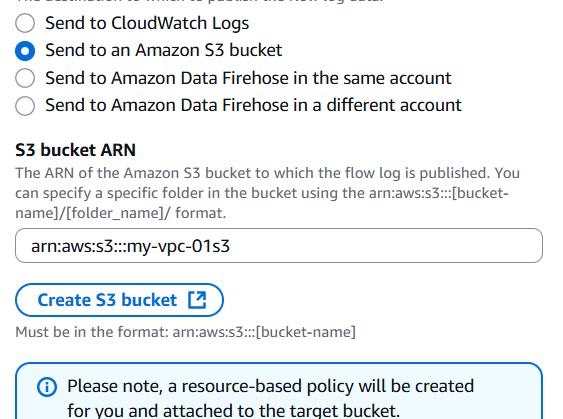
9 . Install Apache Tomcat in private EC2 and deploy a sample app

* Connect to amazon linux
* Download
* Apachetomcat
* Go to cd /opt/tomcat/bin
* Downloafd sample WAR file
* wget <https://tomcat.apache.org/tomcat-10.1-doc/appdev/sample/sample.war>
* 
* Now deploy WAR file
* sudo cp sample.war /opt/tomcat/webapps/
* cd /opt/tomcat/bin
* ls
* 

10.

* Go aws console
* Merge s3 bucket with vpc
* First create s3 bucket
* 
* Now select your vpc , click actions select create flow log



* Filter - all
* Copy s3 bucket arn and paste in flow log
* 
* Select option -send to amazon s3 bucket
* 
* Add text format
* Partition logs – every 24 hr
* Then create a flow log
* 