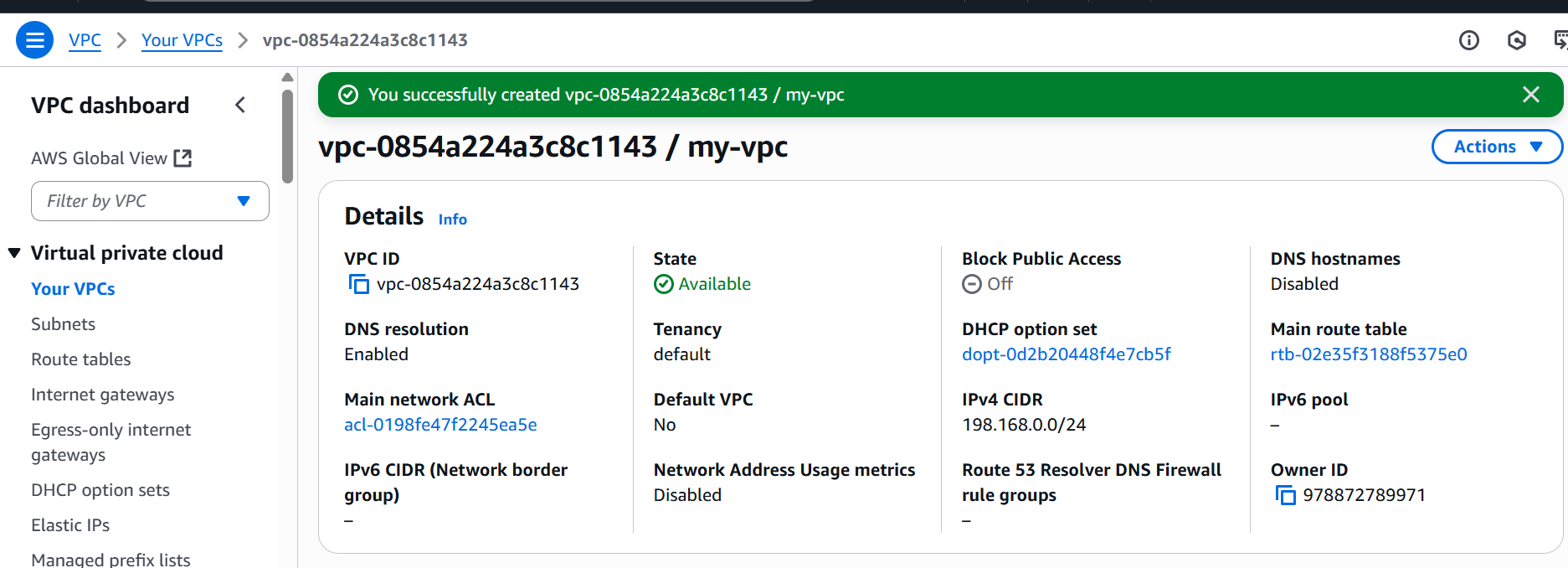
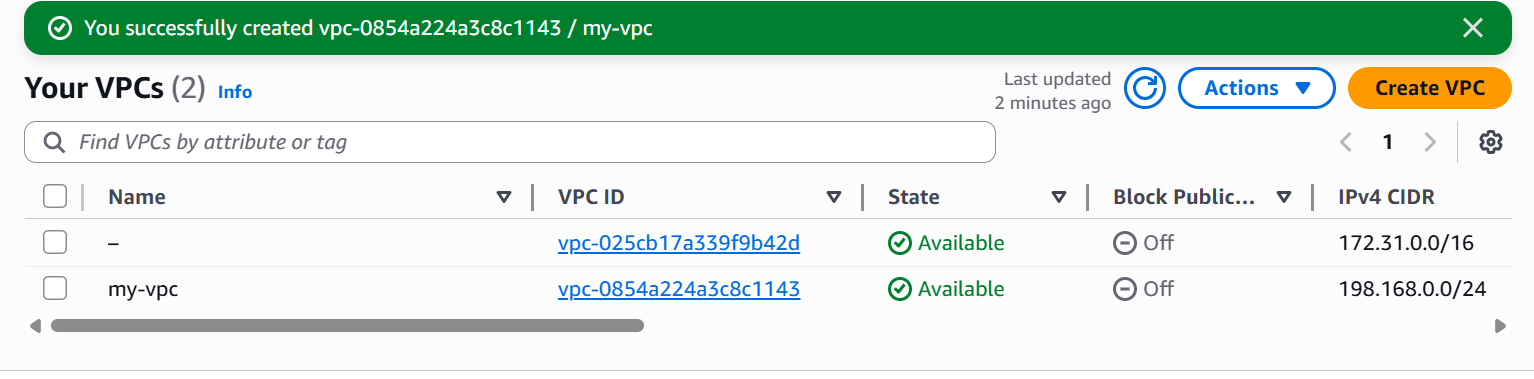
**VPC DAILY TASKS ASSIGNMENT-1**

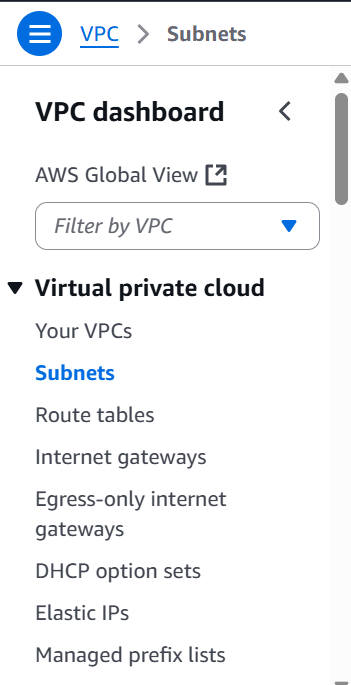
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

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

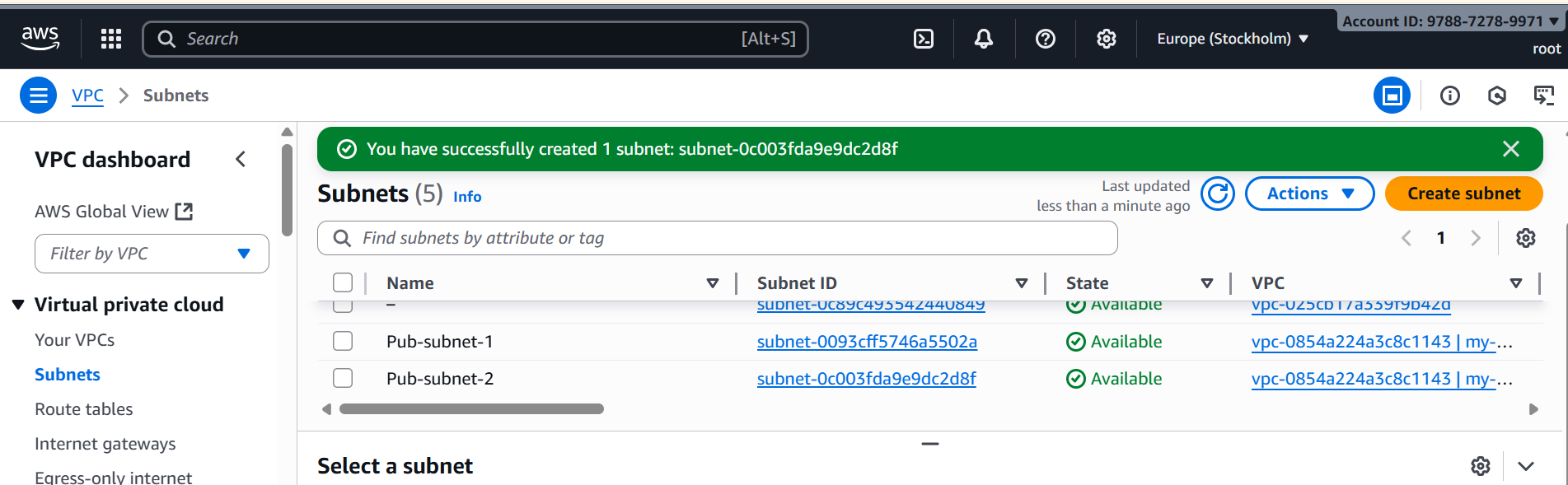
* Go to Aws console and in Search bar search vpc
* Click on vpc
* And click on create vpc
* And give name and
* Select ipv4 CIDR block
* And click on create vpc



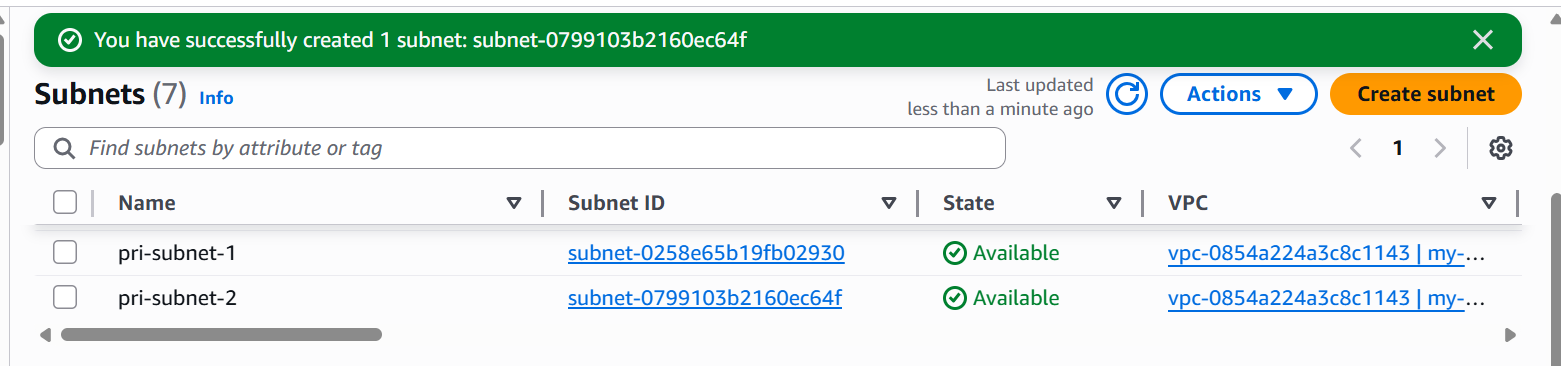




* Here option called subnets ,Click on subnets
* And click on create subnet
* Select vpc (which vpc)
* Create subnet name (pub-sub)
* Click on IPV4 vpc CIDR block and select ip
* And give subnet CIDR block
* And click on create subnet
* Created 2 public subnets

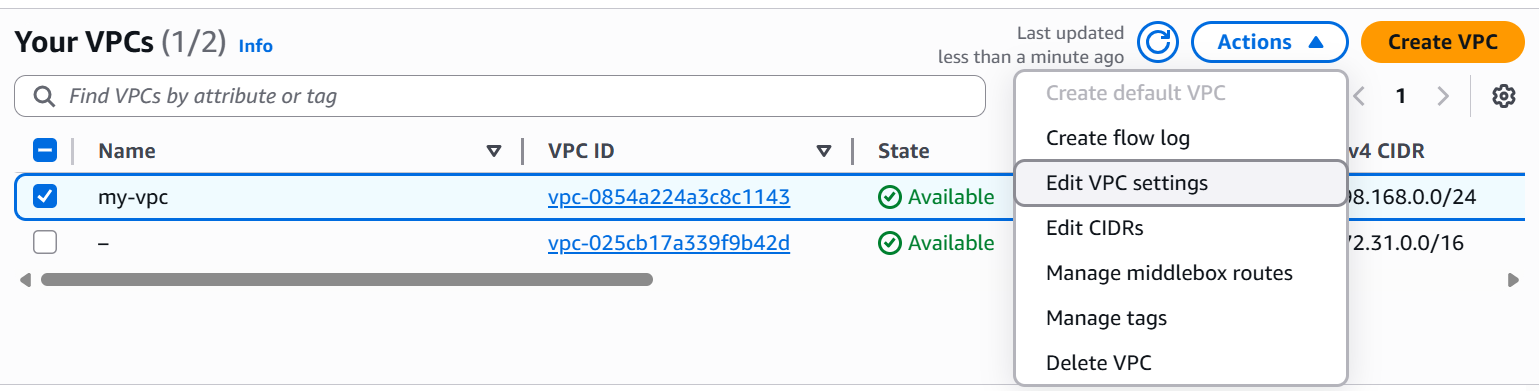


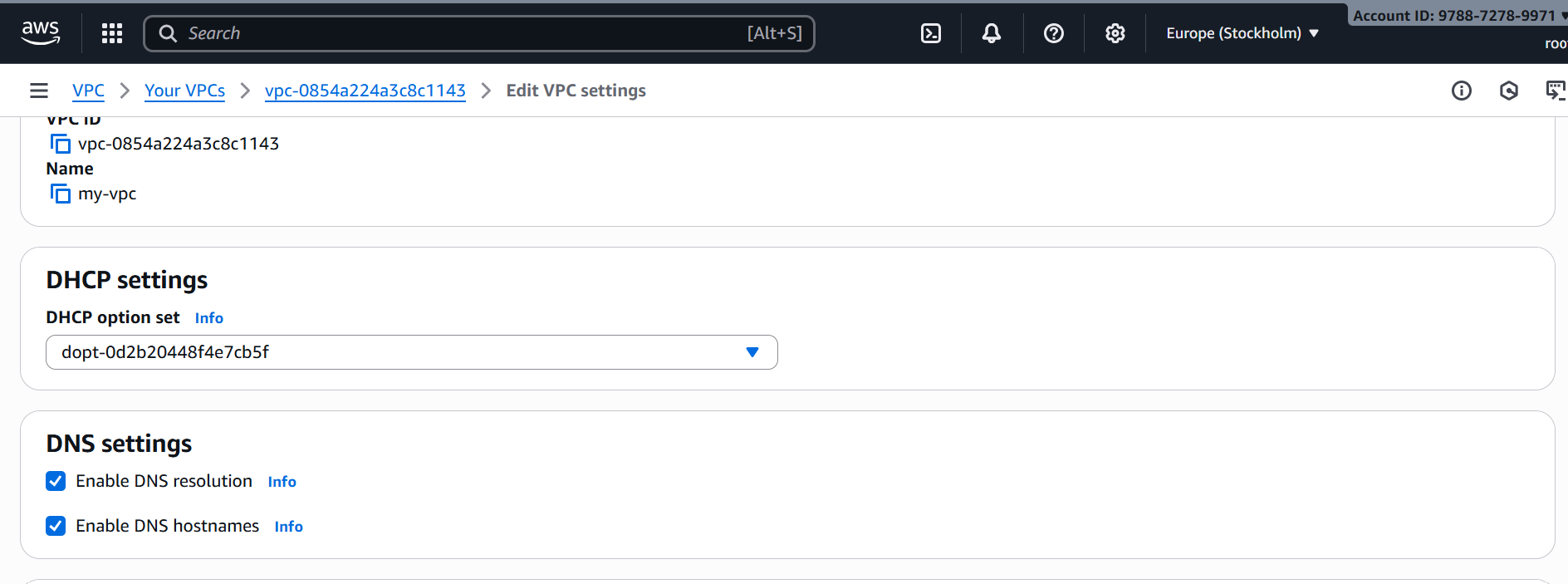
* Create 2 private subnets



**2. Enable DNS Hostname in VPC**

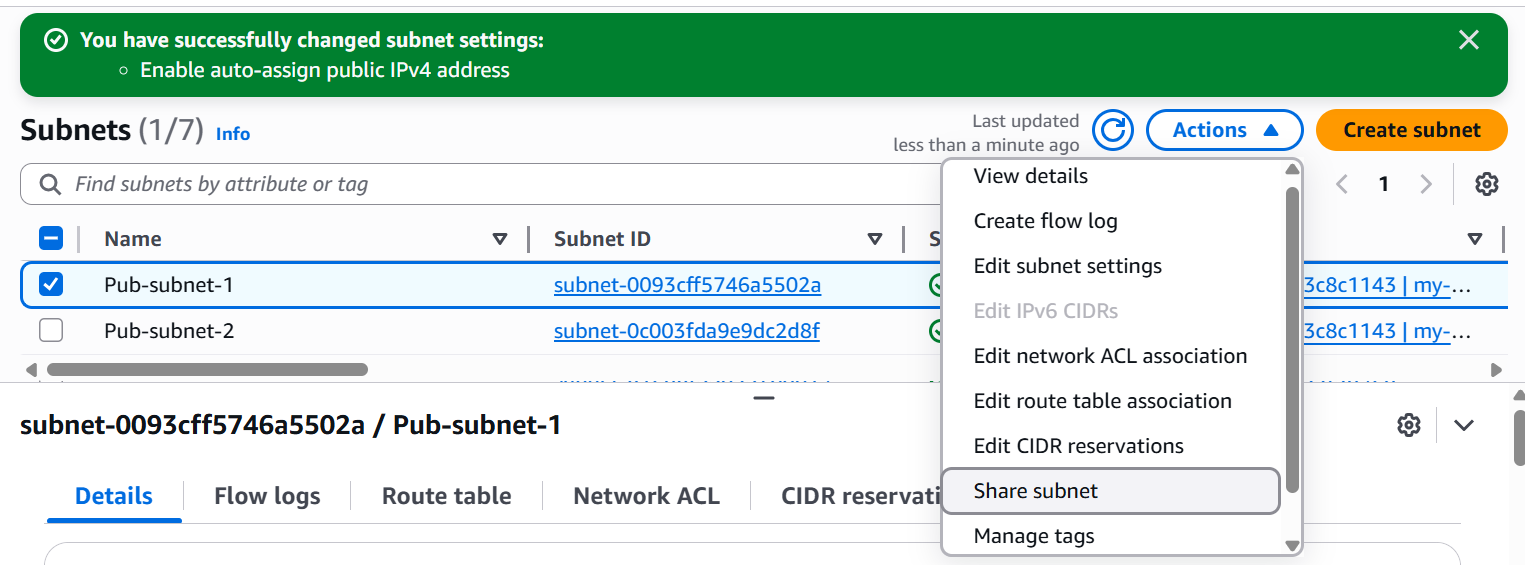
* Go to **VPC Dashboard** → **Your VPCs**.
* Select the VPC you created.
* In the **Actions** menu, click **Edit VPC settings**.
* Enable the checkbox **DNS hostnames**.
* Save changes

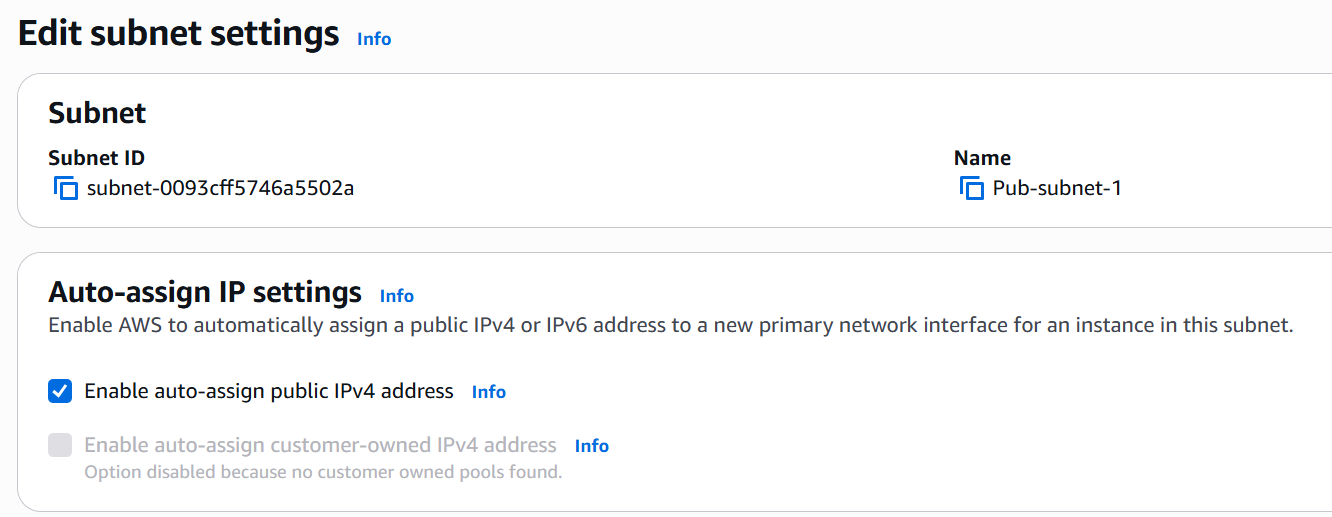


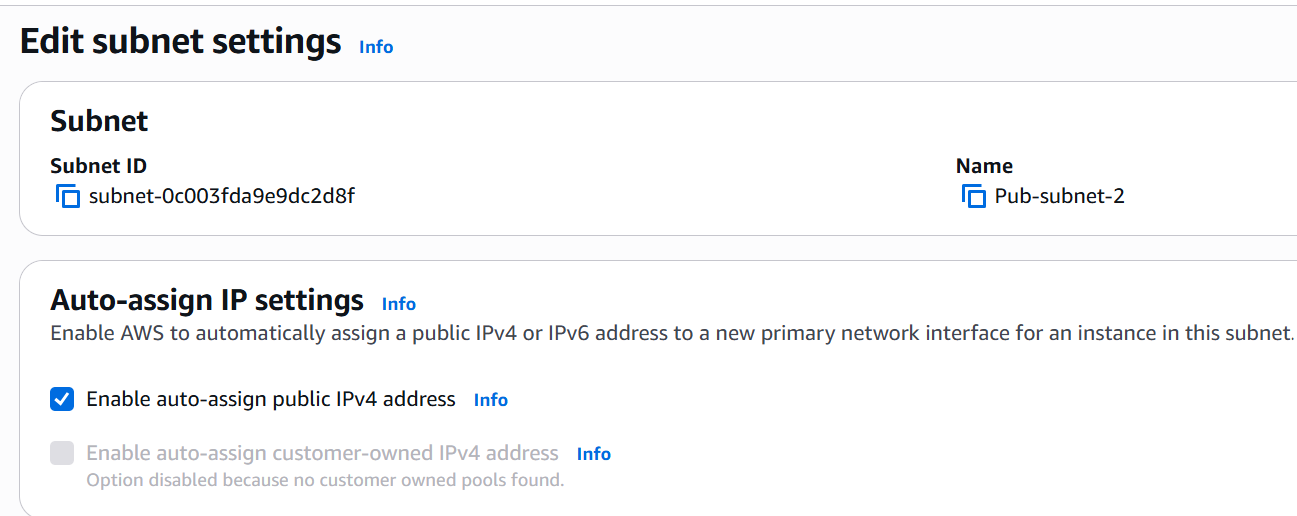


**3. Enable Auto Assign Public IP in 2 public subnets**

* Go to **VPC Dashboard , click on Subnets**
* Select your **Public Subnet**
* Click **Actions → Edit subnet settings**
* Enable **Auto-assign IP settings → Auto-assign public IPv4 address**
* Save changes
* Repeat the same steps for your **second public subnet**

****

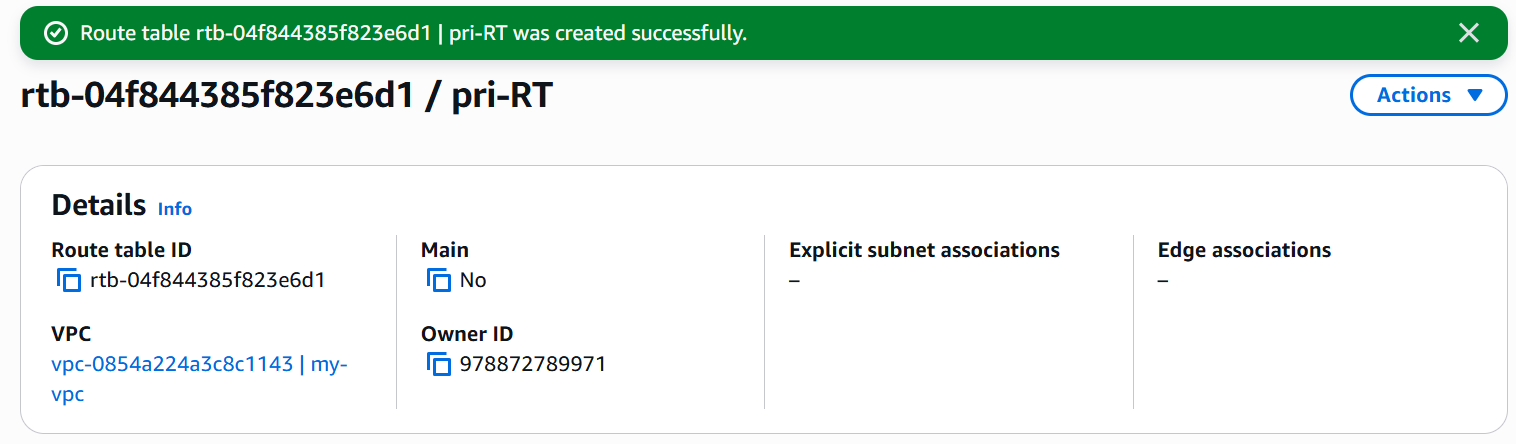




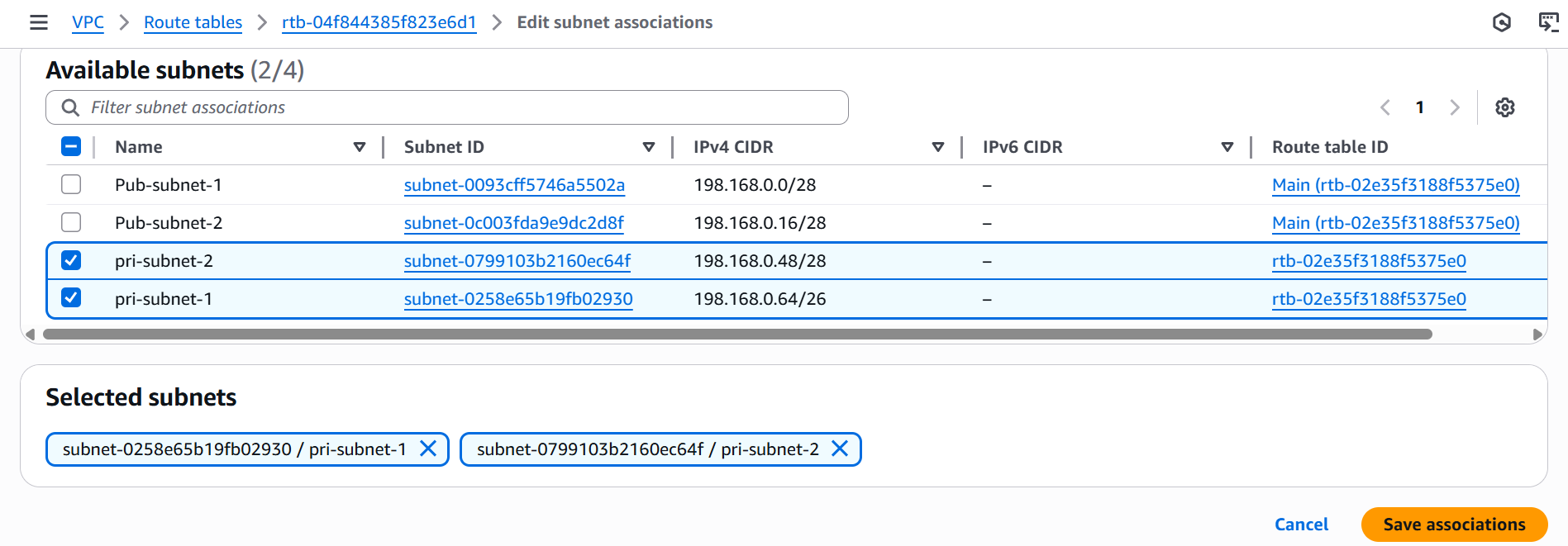


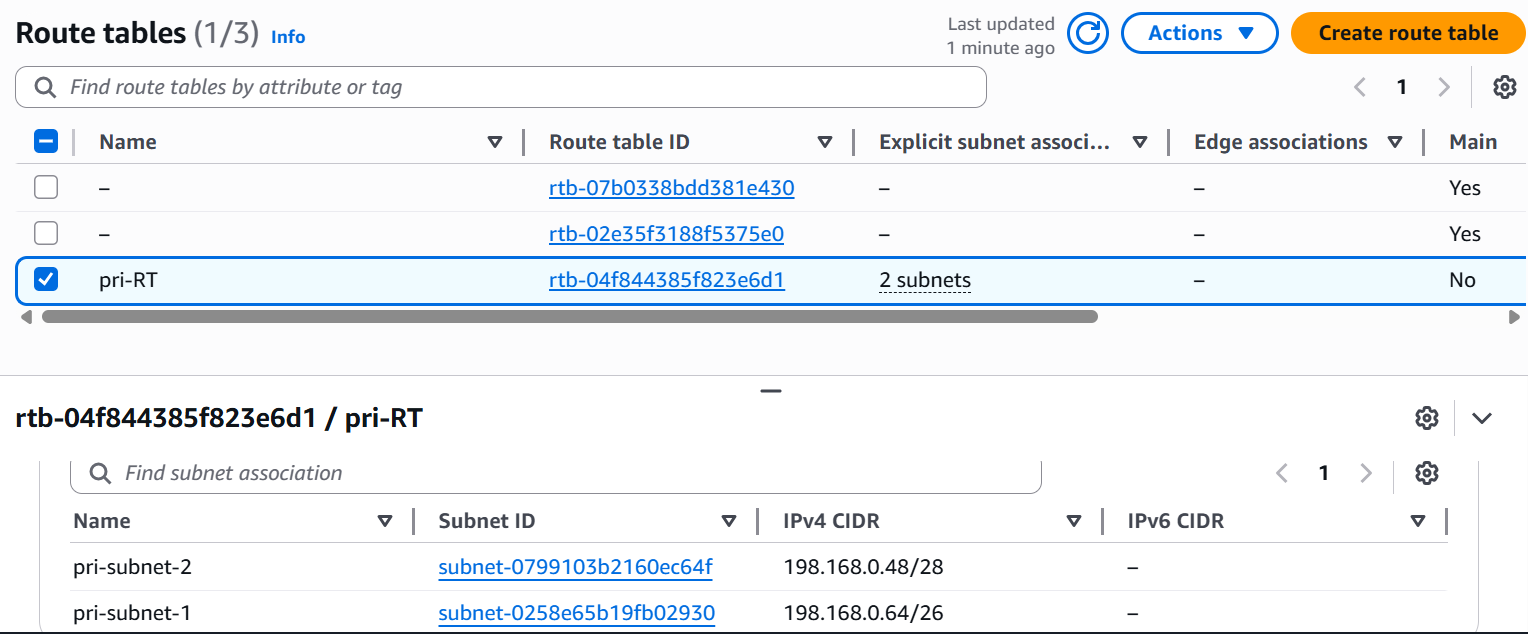
**4. Add 2 private subnets in private route table**

* Go to **VPC →**click on **Route Tables →** click on **Create route table**
* Give name like **Pri-RT**
* Attach it to the **my-VPC**



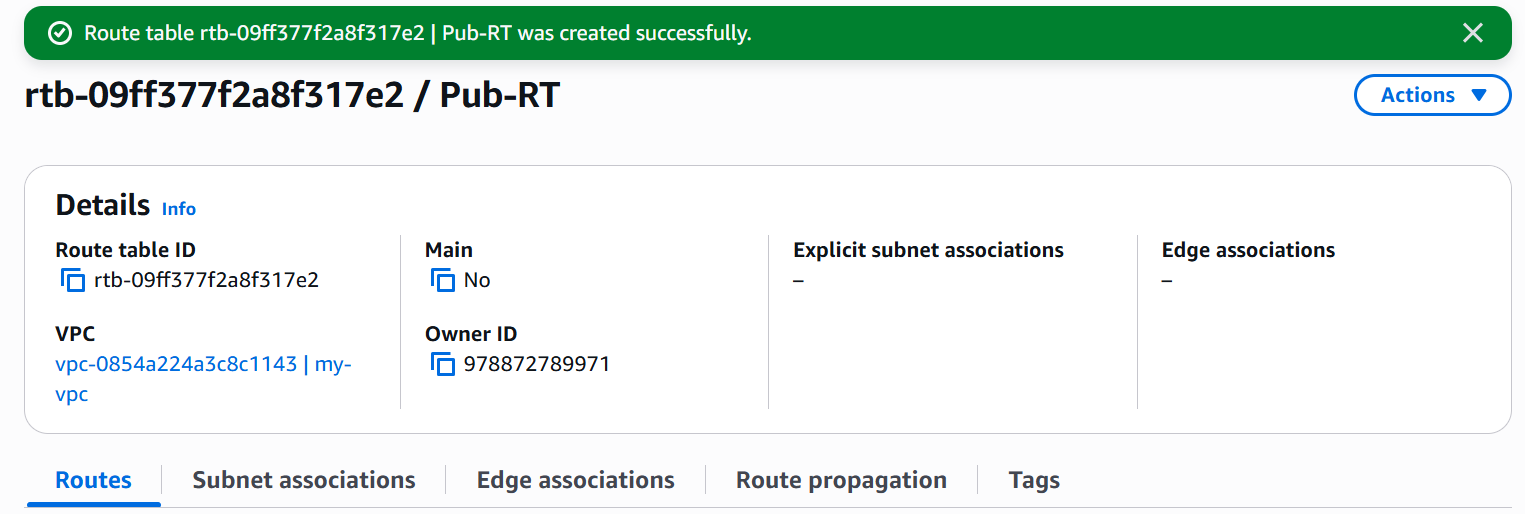
* **Add Private Subnets to the Route Table**
* Select your new **Private Route Table**
* Go to **Subnet associations** → **Edit subnet associations**
* Select the **two private subnets** you created
* Save



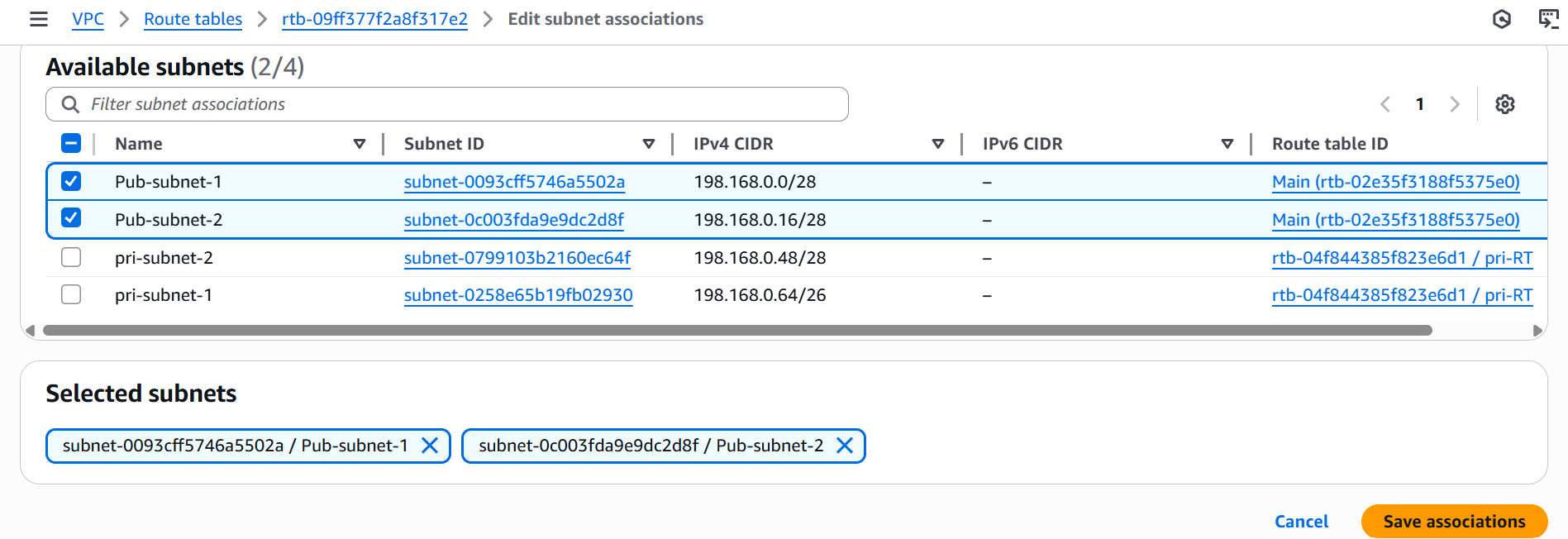


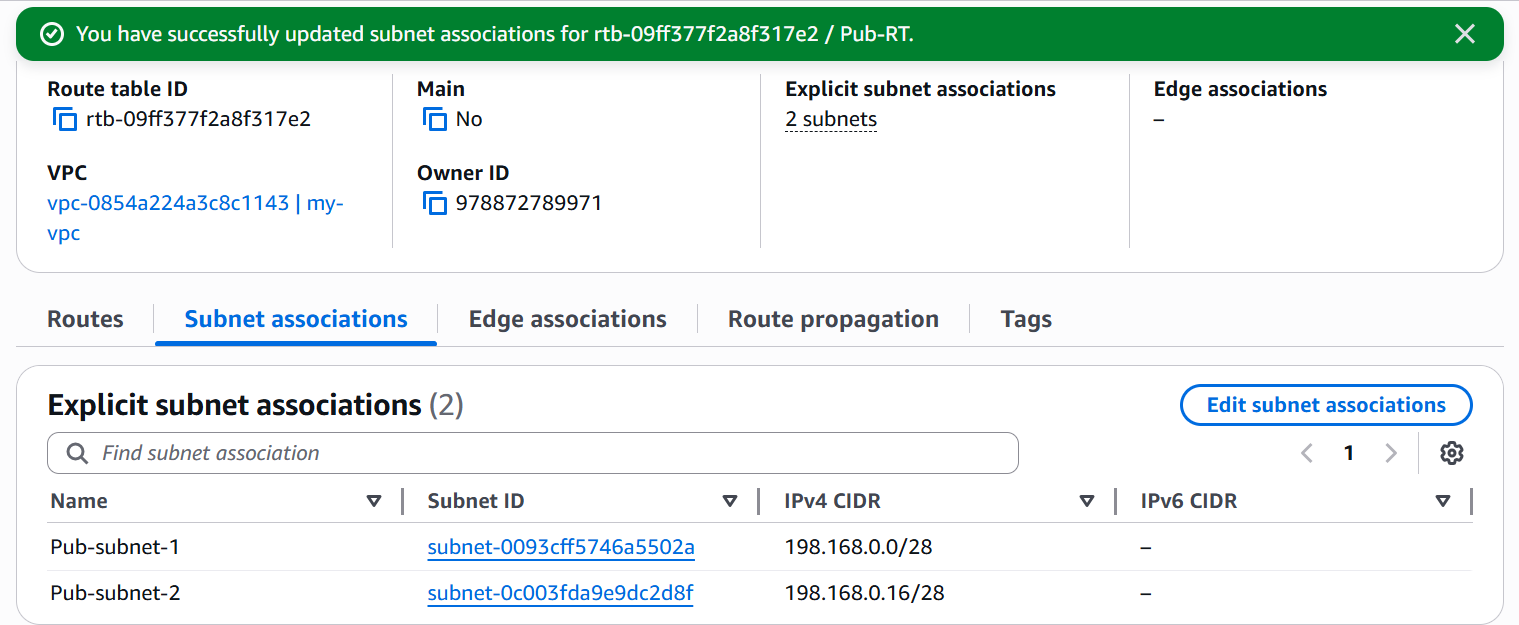
**5. Add 2 public subnets in public route table**

* Go to **VPC →**click on **Route Tables →** click on **Create route table**
* Give name like **Pub-RT**
* Attach it to the **my-VPC**



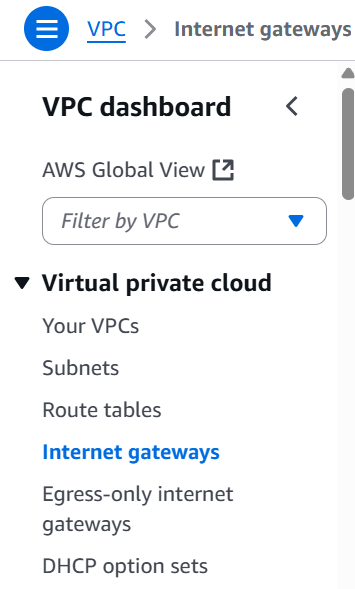
* **Add public Subnets to the Route Table**
* Select your new **Public Route Table**
* Go to **Subnet associations** → **Edit subnet associations**
* Select the **two public subnets** you created
* Save



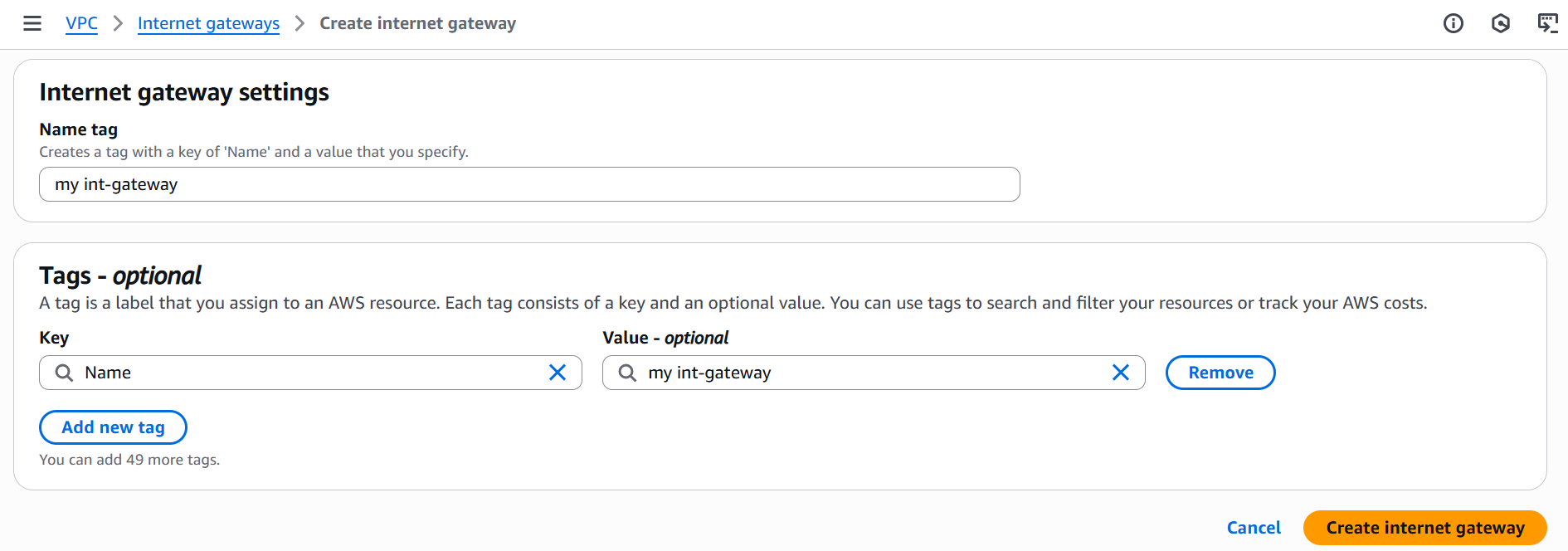


**6. Public route table will have the routes to internet and local**

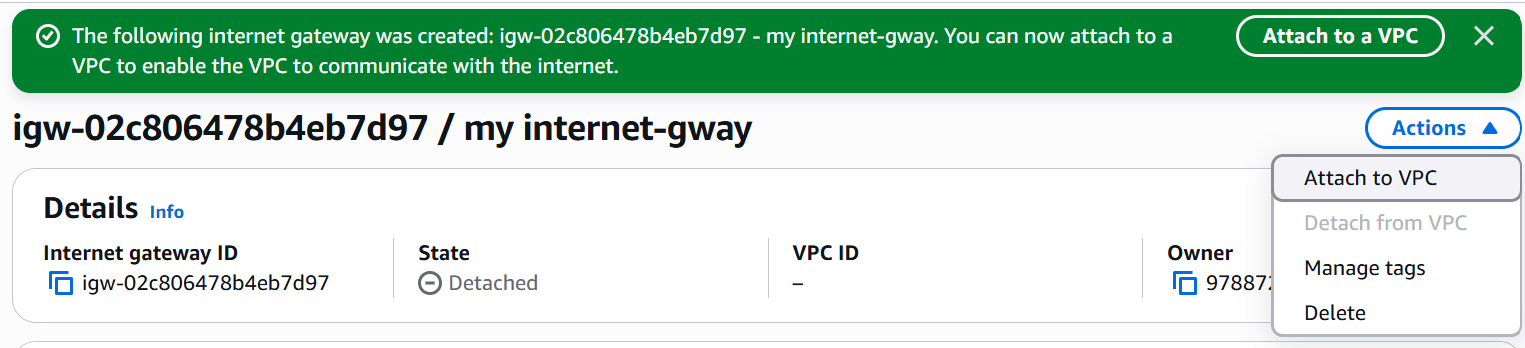
* Go to VPC click on Internet Gateways

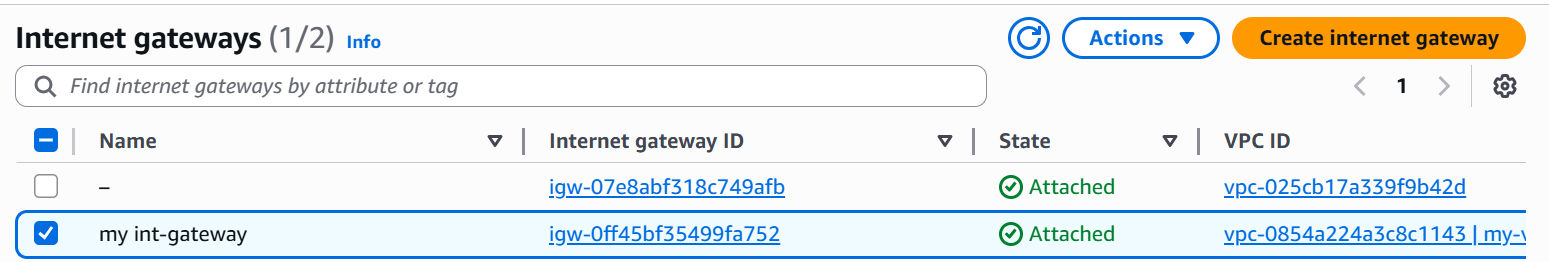


* Create internet gateway and name it
* Select the created IGW

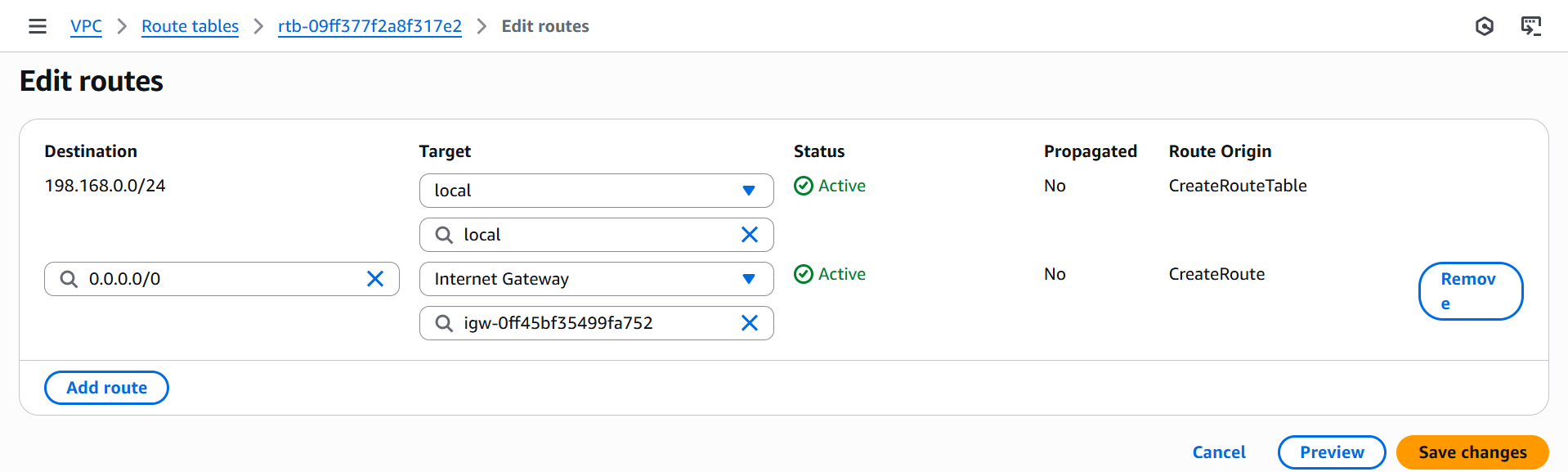


* Click on Actions
* Attach to VPC and select your VPC

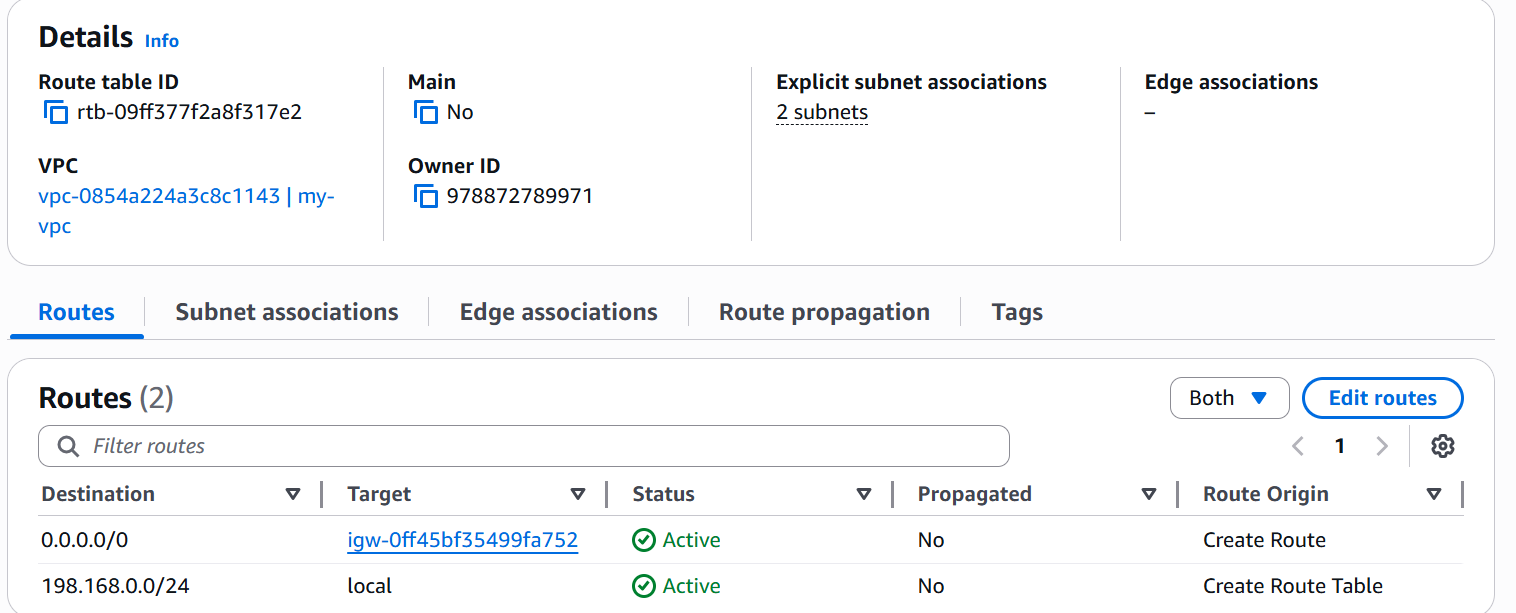




* + Go to route table and click on rotes
  + Click on edit routes and add route
  + Give destination as: **0.0.0.0/0**
  + Give target : choose your **internet gateway (igw - xxxx)** and save

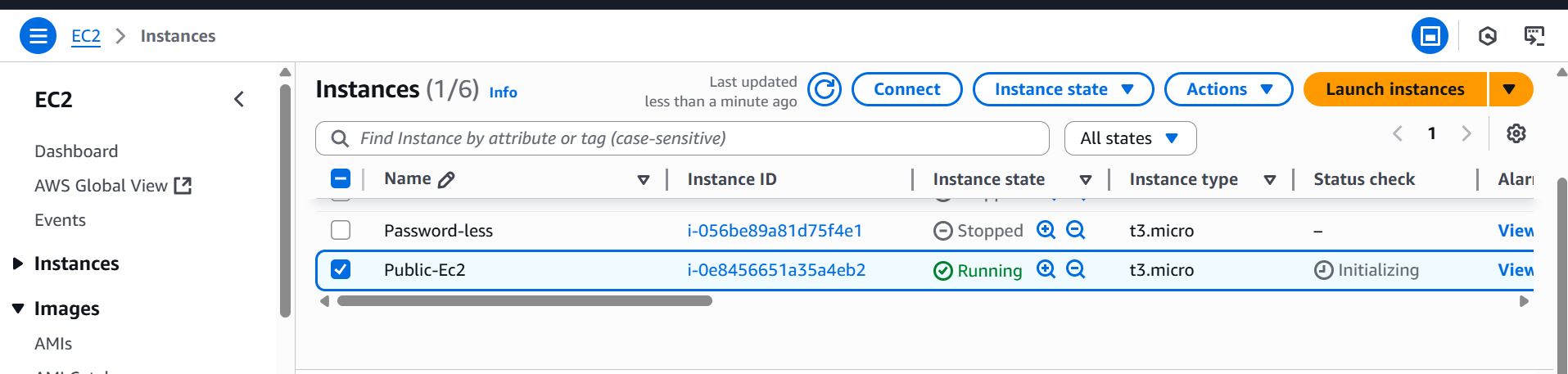


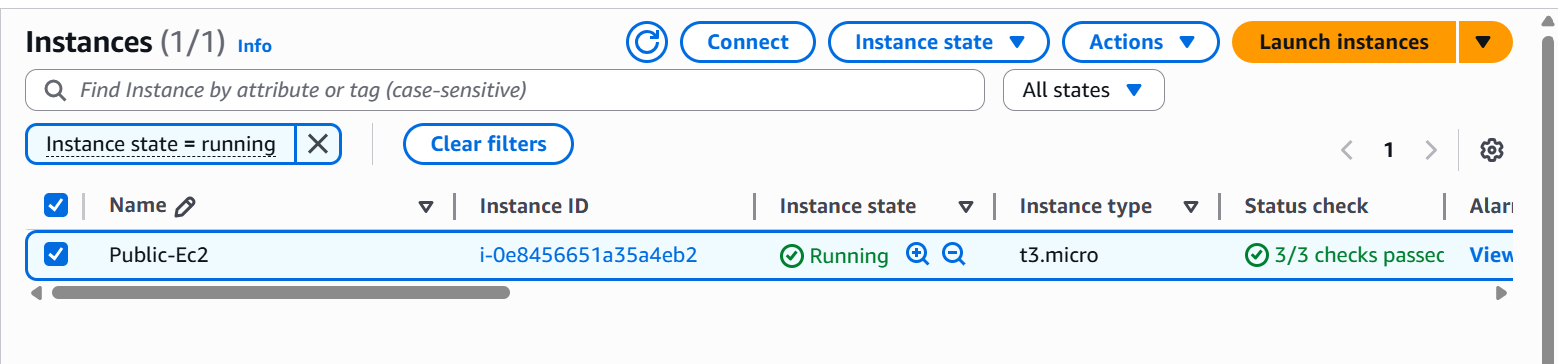
* + Public route table will have the routes to internet and local



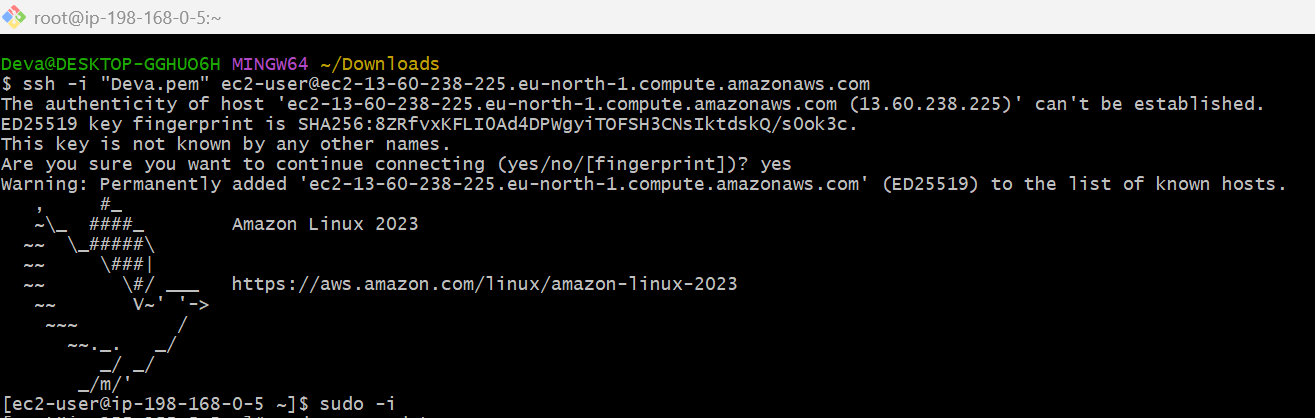
**7. Create EC2 in public subnet with t2.micro and install PHP**

* Go to **EC2 → Click on Instances**
* **Click on Launch Instance**
* Give name **Public-EC2**
* **Select AMI** Amazon Linux 2
* **Instance type**: t3.micro (Free tier eligible).
* **Key Pair -** Create or use an existing one.
* **Click on Network settings**:
* VPC: Select your VPC.
* Subnet: Select your **Public Subnet**.
* Auto-assign Public IP: Enable
* **Security Group** - select existing one and give default security group.
* **Click on launch instance**

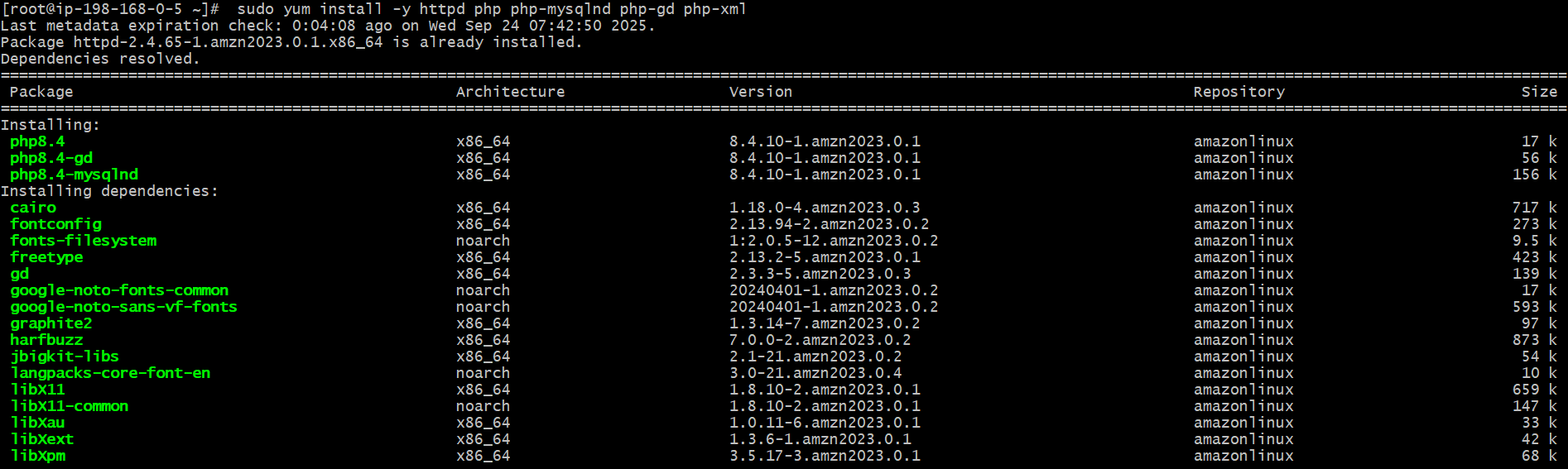
****

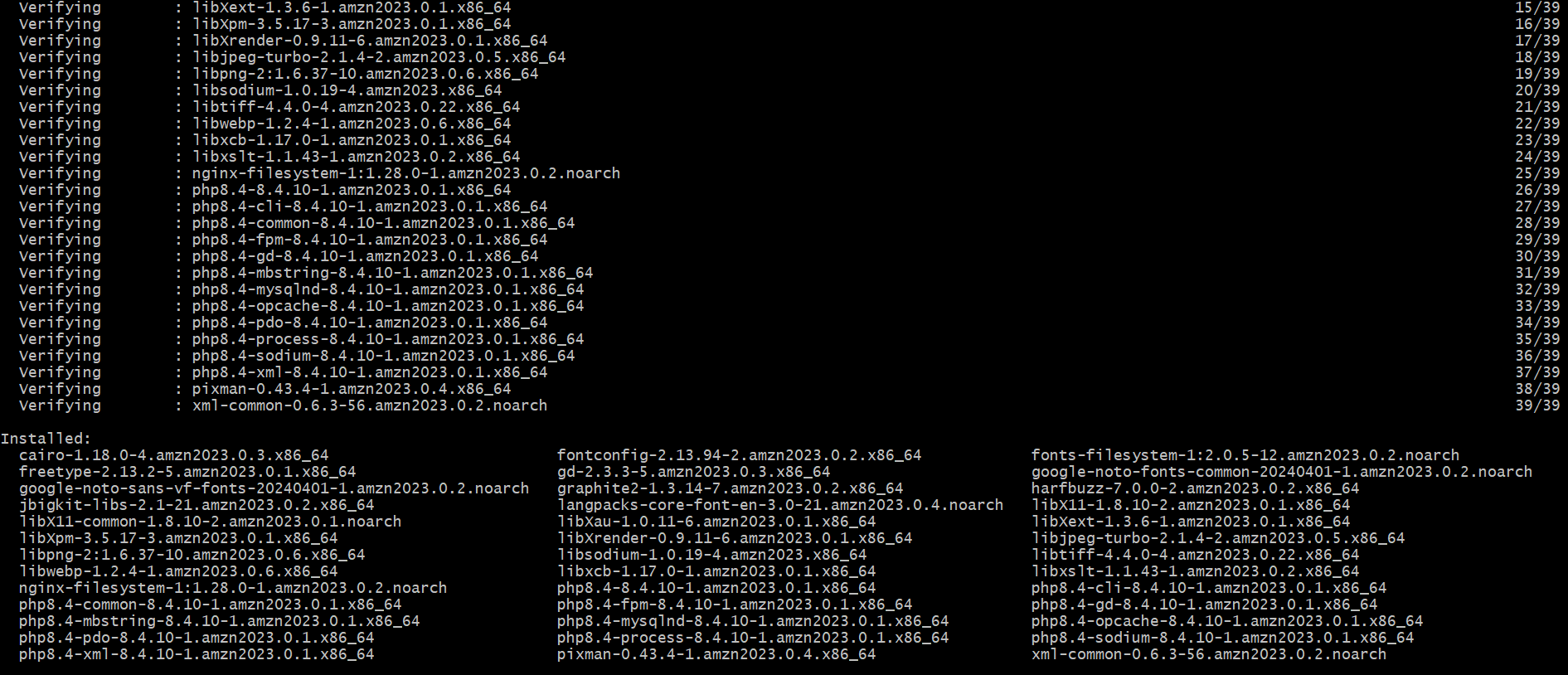
****

* + After instance is running, click **Connect → SSH**

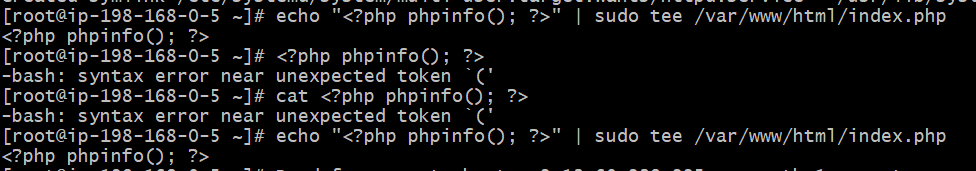


* + **Install php**

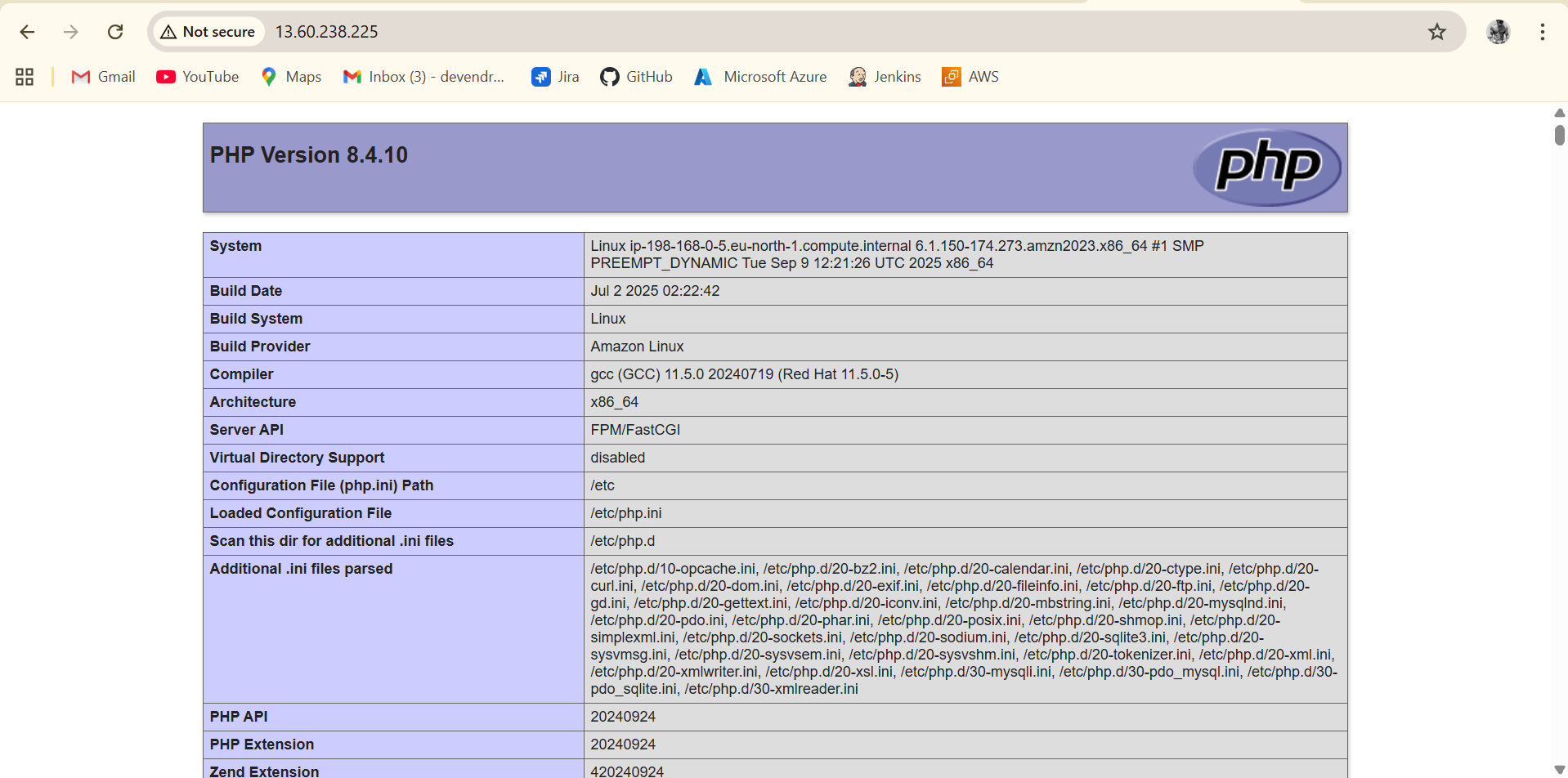
****

****

* Create a sample **php** file

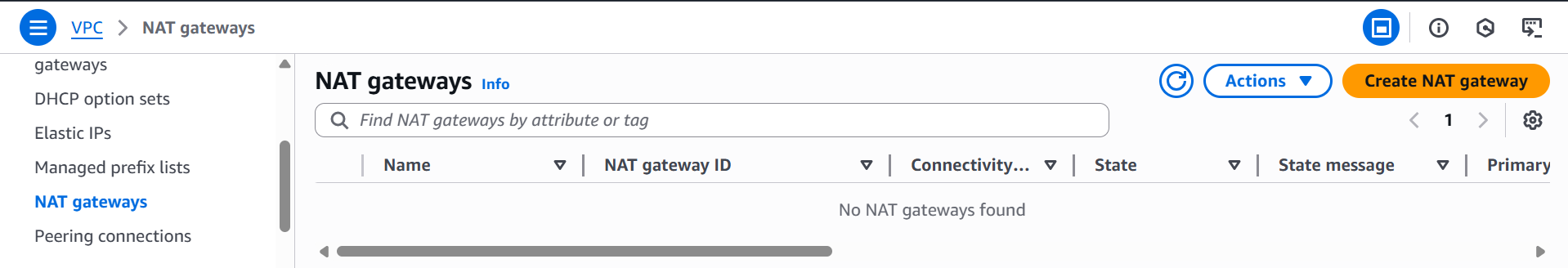
****

* Open browser and check [**http://(Public-IP)**](http://(Public-IP))you should see PHP info page

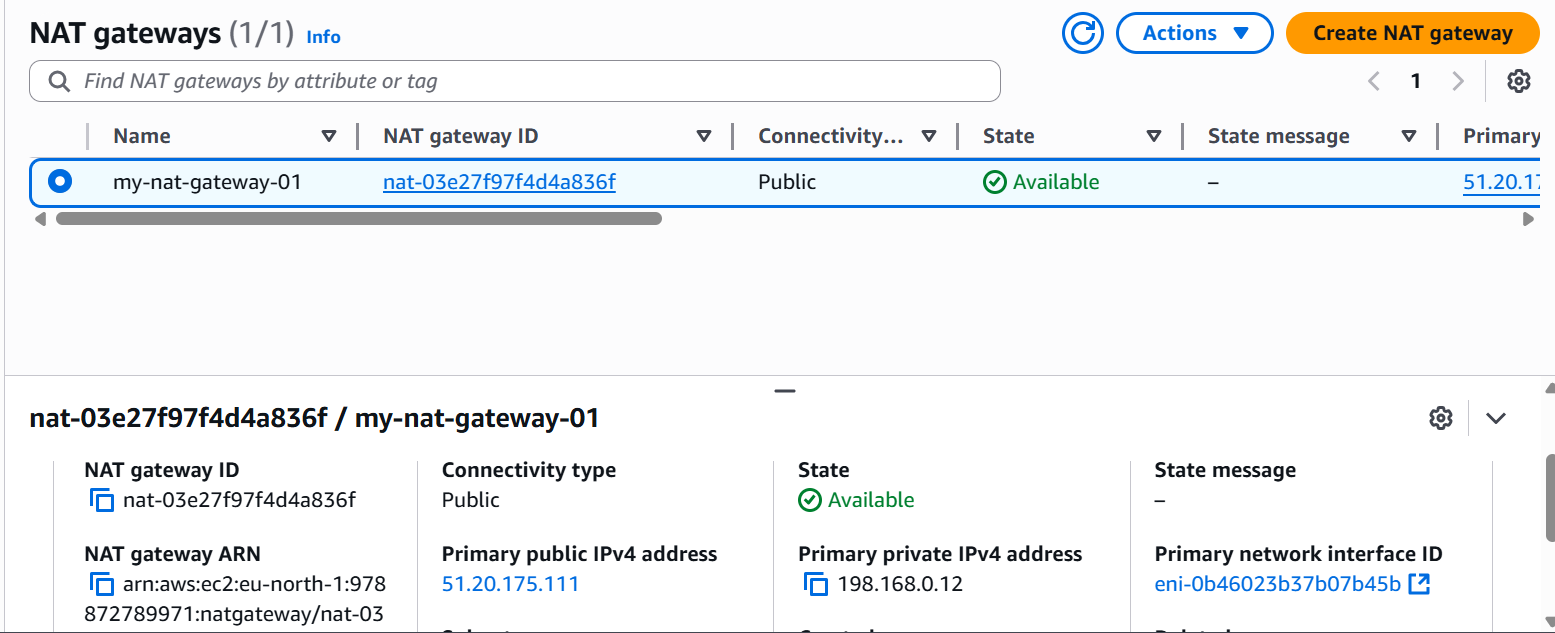
****

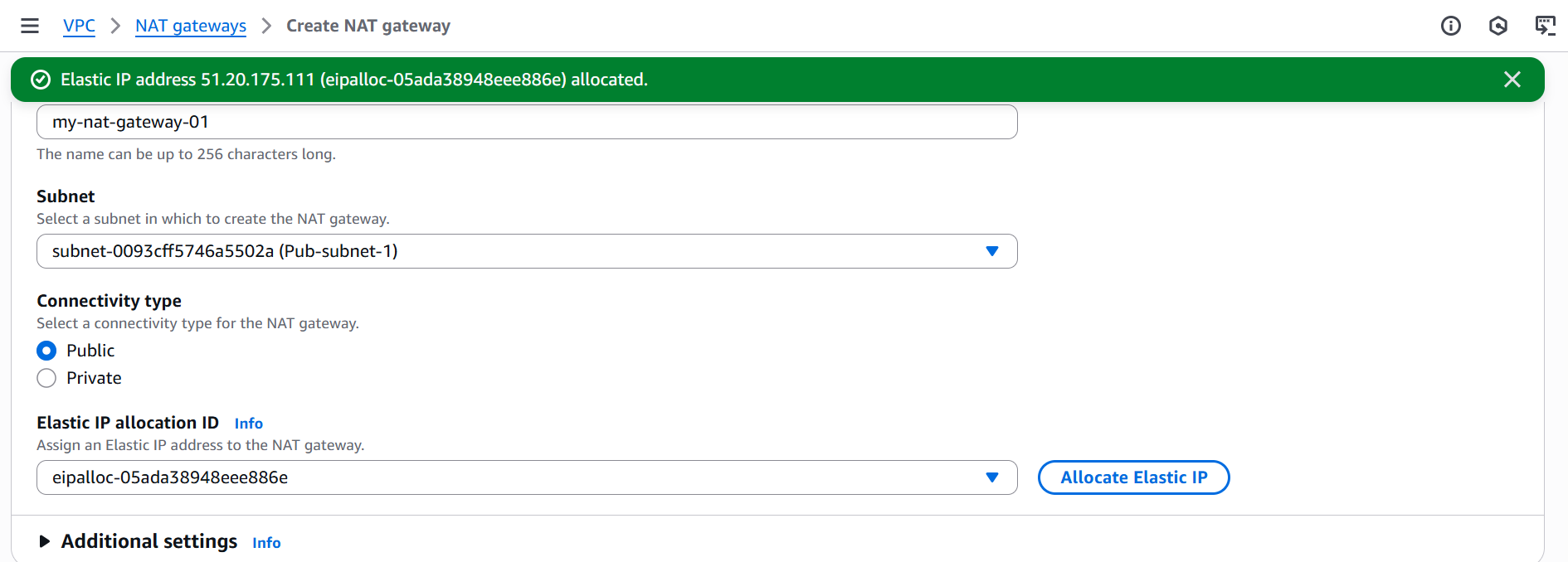
**8. Configure NAT gateway in public subnet and connect to private instance.**

* Go to **VPC**
* Click on **NAT Gateways**
* **Click on Create NAT Gateway**

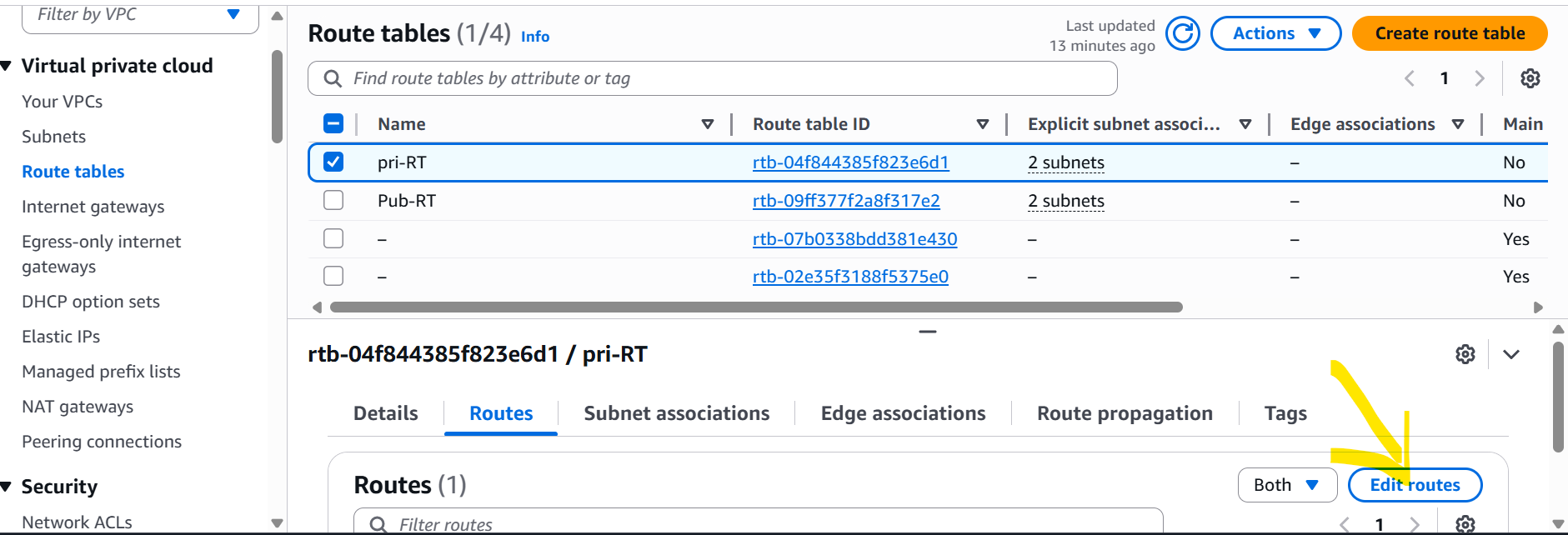


* **Subnet**: Select one of your **Public Subnets** (because NAT needs internet access)
* **Elastic IP** → Allocate a new Elastic IP.
* Create NAT Gateway - wait until status is **Available**

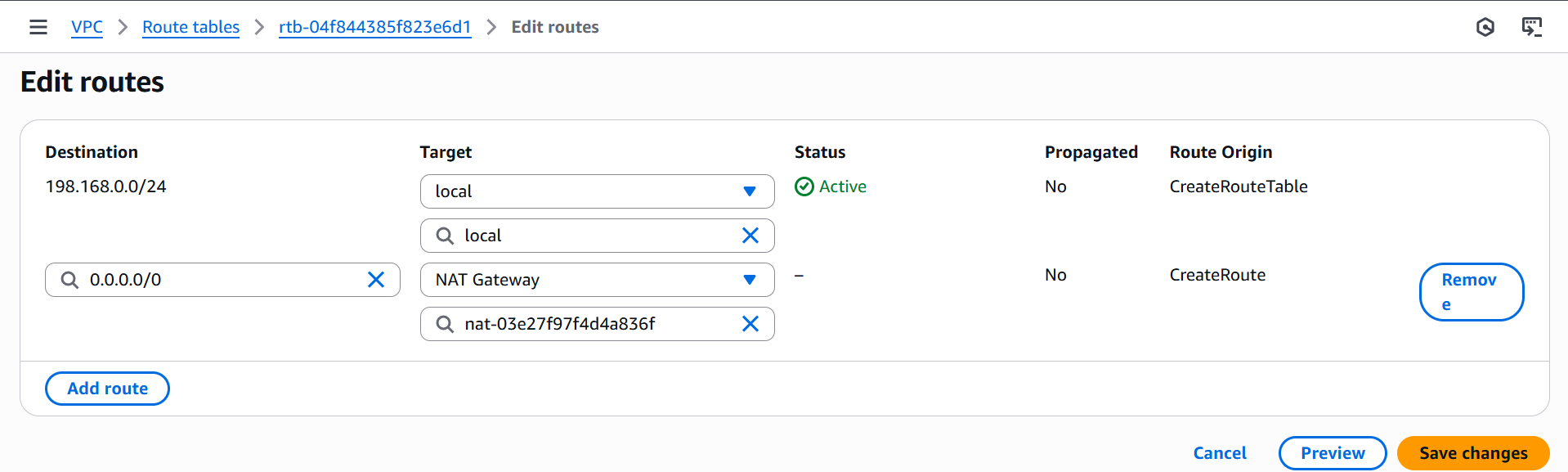


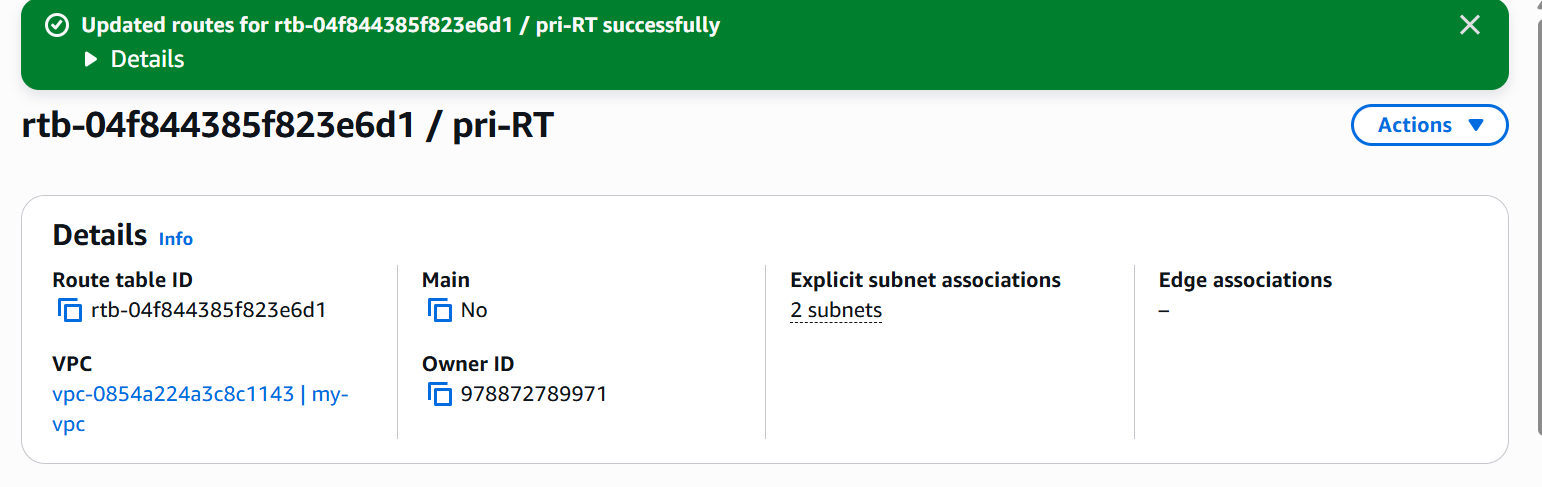


* Go to **Route Tables and Select your Private Route Table**.
* Click on Routes → Edit Routes → Add routes

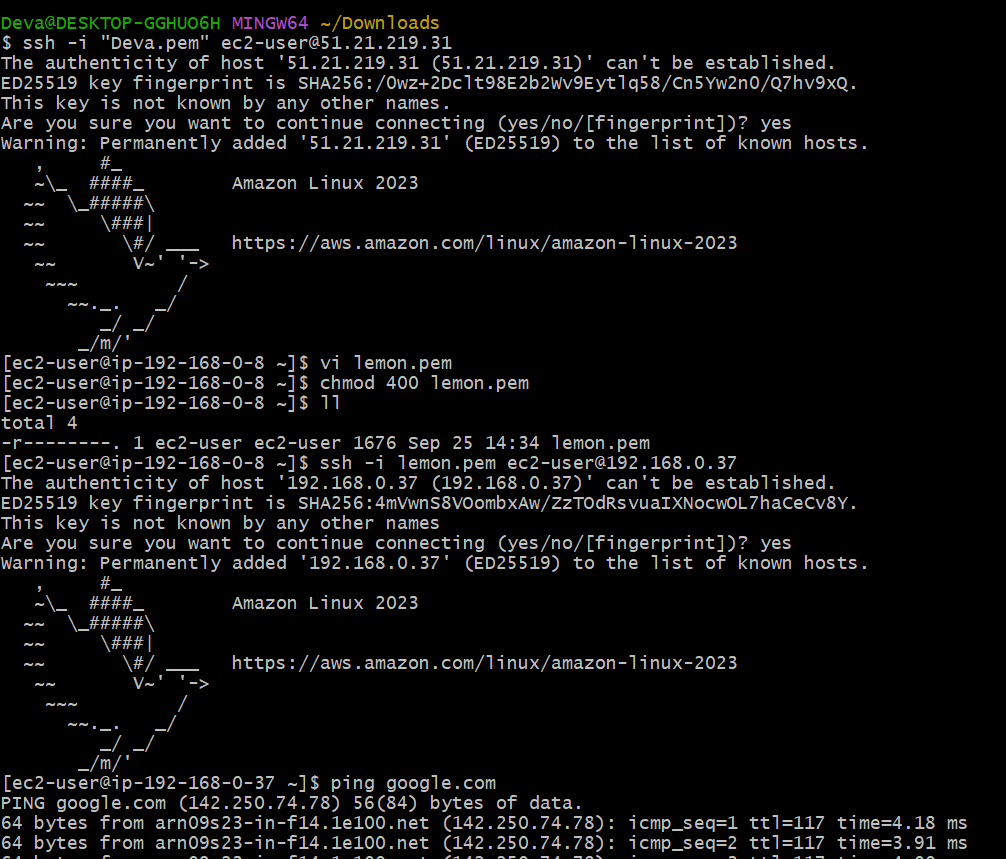


* **Give Destination:** 0.0.0.0/0
* **Target:** NAT Gateway ID (nat-xxxxxxx)
  + Click on save changes



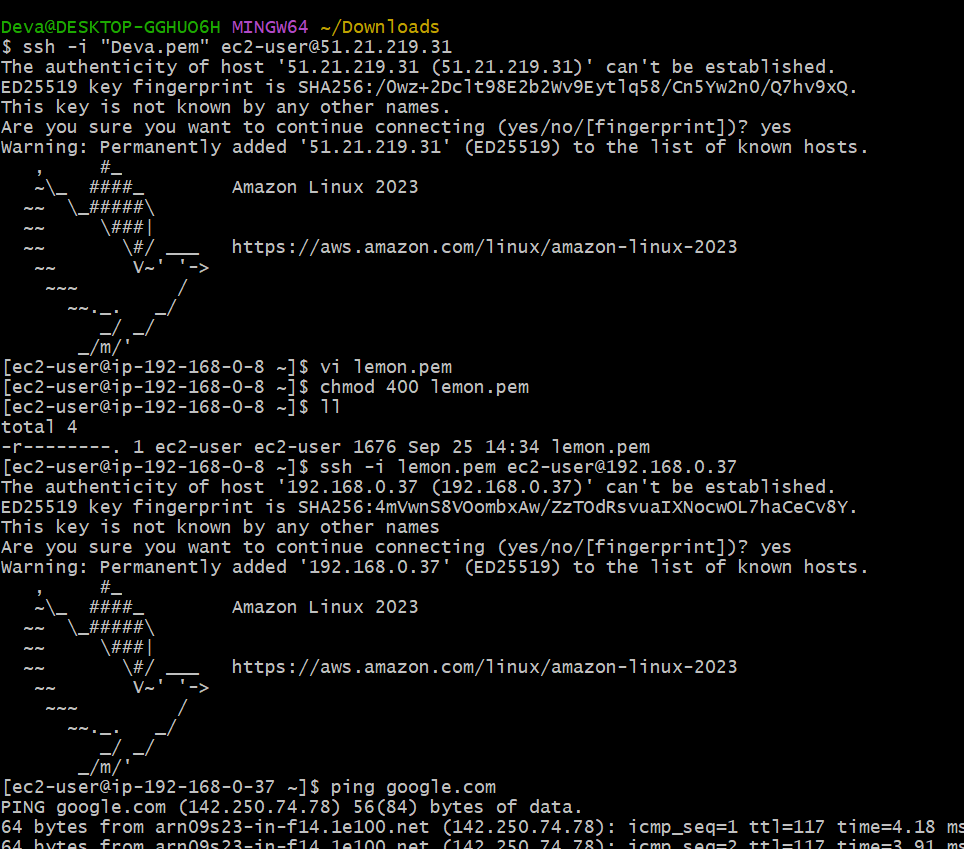


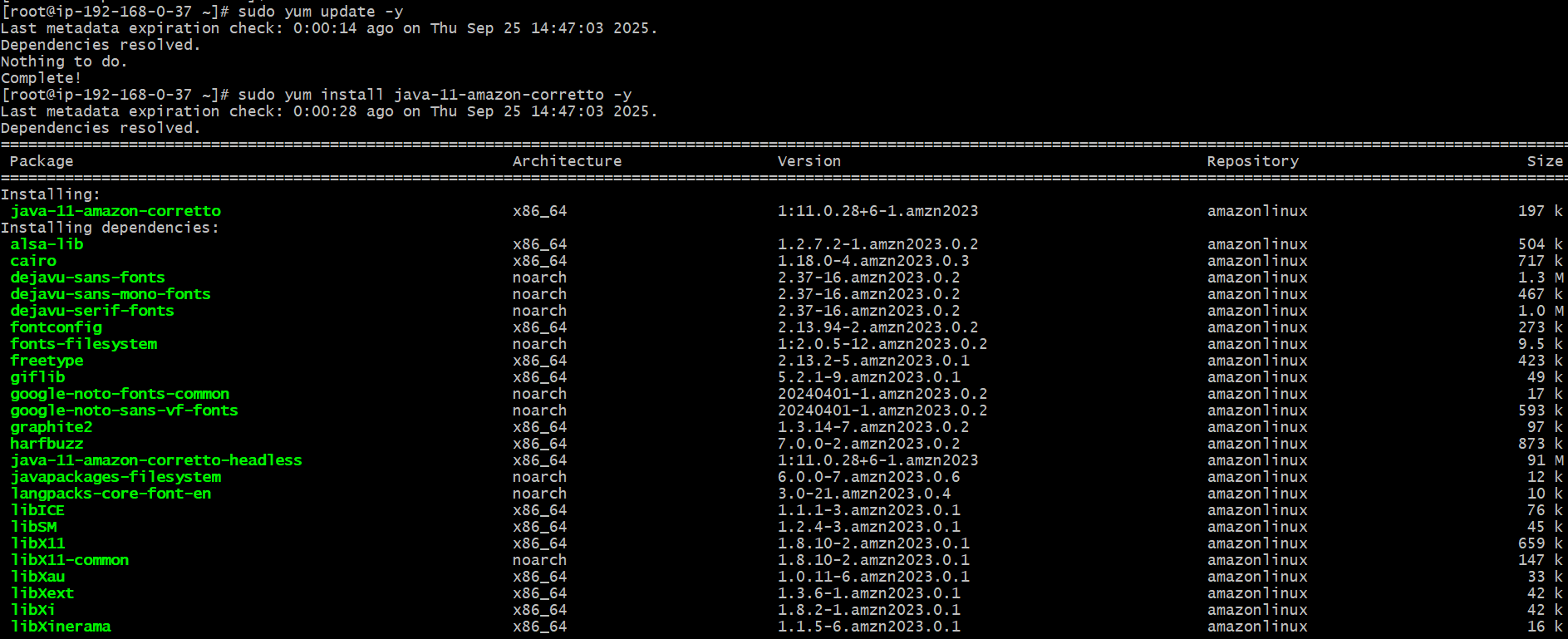
* First, connect to **Public Instance** (via SSH)
* From public instance, SSH into **Private Instance** using its **Private IP]**
* **Result:**
* Public Instance → direct internet via IGW.
* Private Instance → Internet via **NAT Gateway**.
* You can connect to the Private Instance **through the Public Instance (bastion)**.



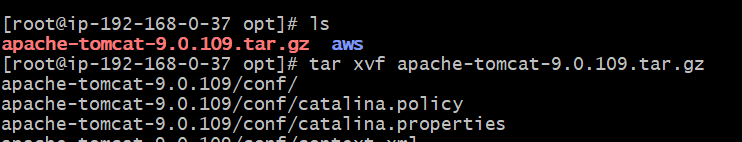
**9. Install Apache Tomcat in private EC2 and deploy a sample app**

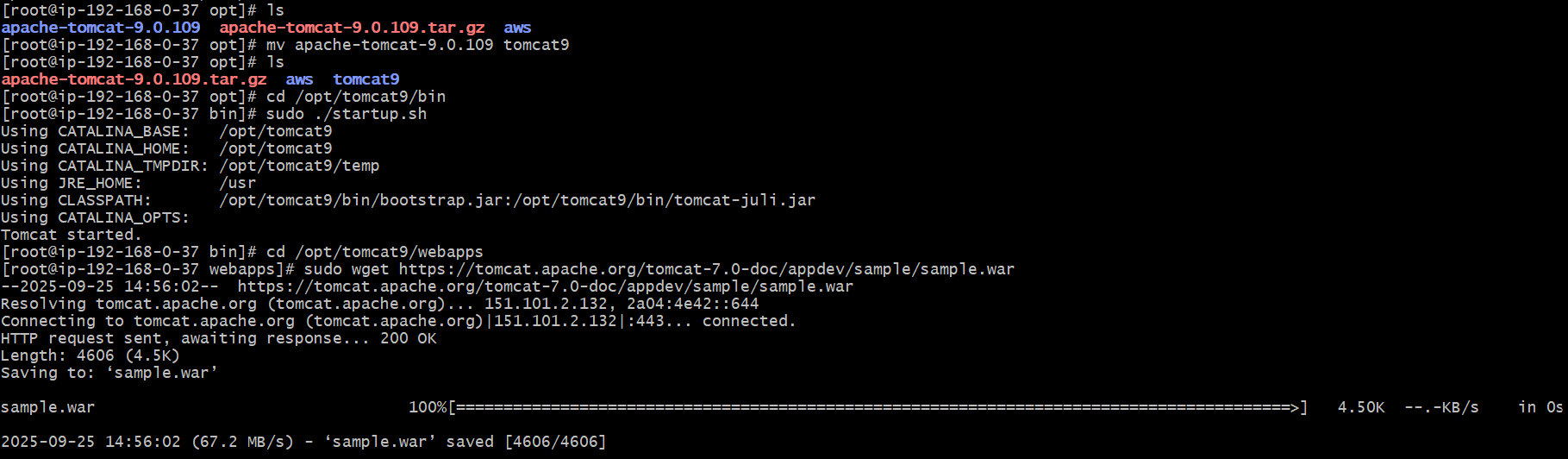
* First, SSH into **Public EC2** (Bastion Host)
* **ssh -i mykey.pem ec2-user@<Public-EC2-Public-IP>**
* From Bastion, SSH into **Private EC2**
* **ssh -i mykey.pem ec2-user@<Private-EC2-Private-IP>**
* **Install Java**
* Tomcat needs Java.
* **Install Tomcat:**
* **sudo yum update -y**
* **sudo amazon-linux-extras install java-openjdk11 -y**
* **wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.95/bin/apache-tomcat-9.0.95.tar.gz**
* **tar -xvzf apache-tomcat-9.0.95.tar.gz**
* **mv apache-tomcat-9.0.95 /opt/tomcat**
* **sh /opt/tomcat/bin/startup.sh**

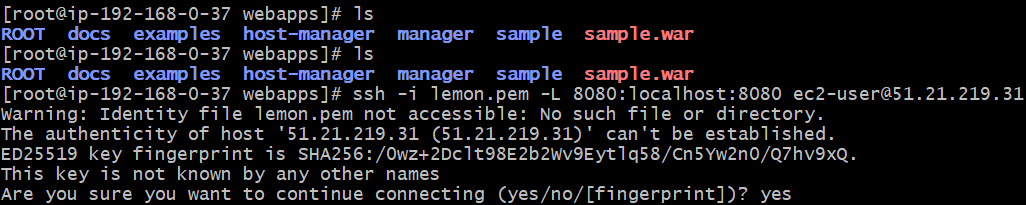
****

****

****

****

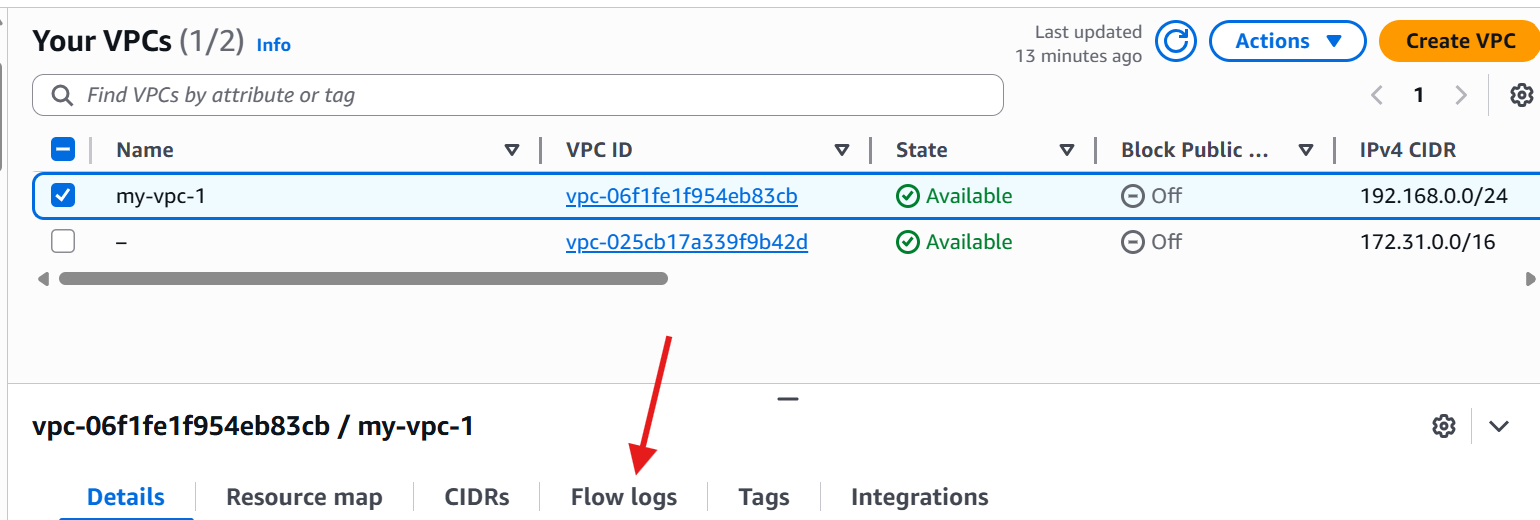
****

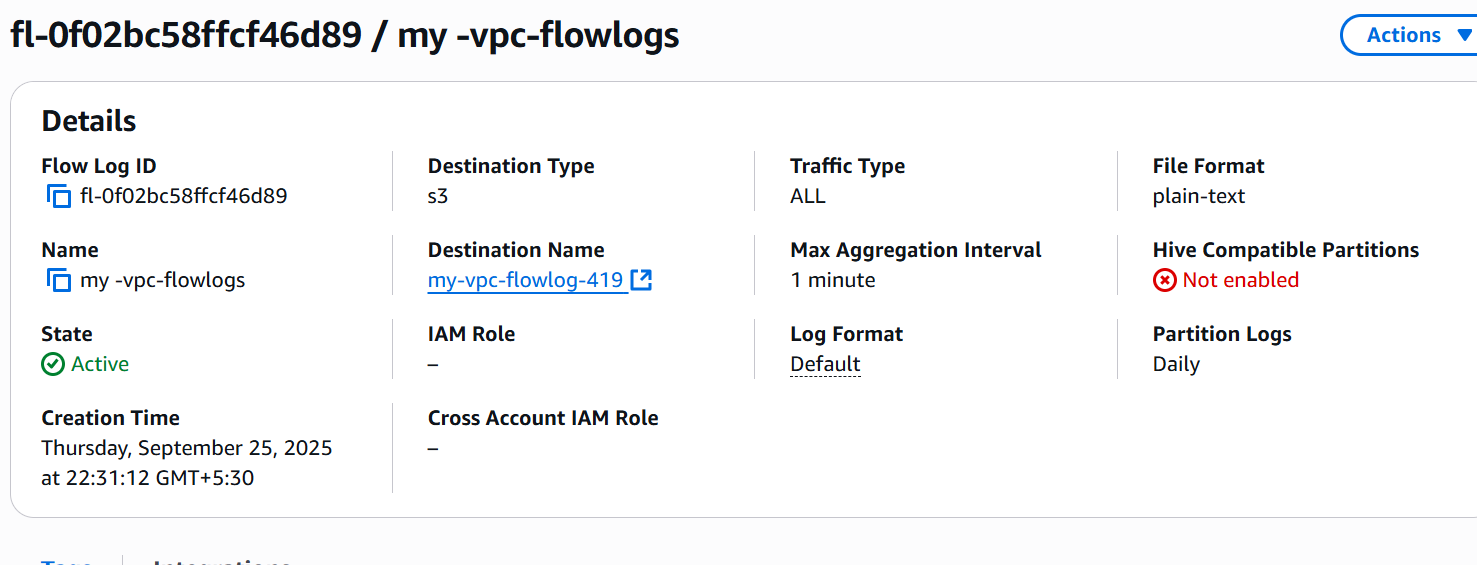
****

**http://localhost:8080/sample**

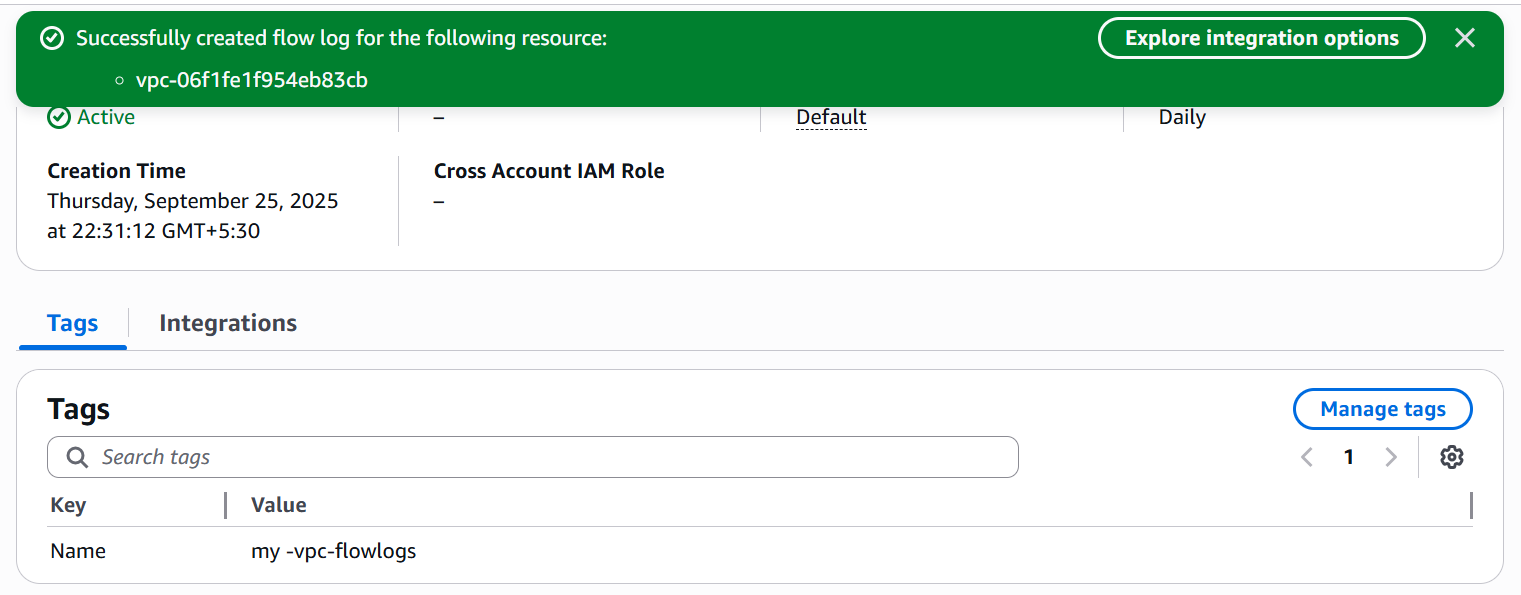
**10. Configure VPC flow logs and store the logs in S3 and Cloud Watch**

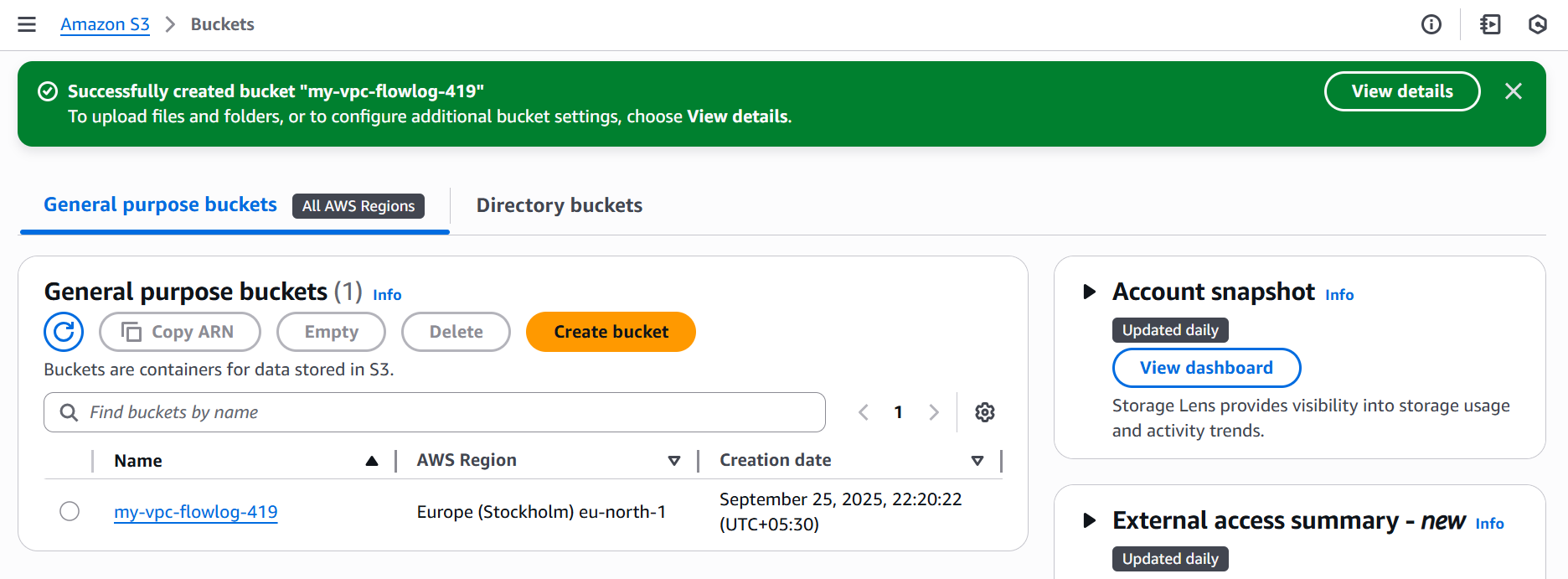
* **S3 flow logs**
* Go to vpc and click on your vpc
* Click create flow log
* Give name like **my flow log**
* Give filter **All**
* And create flow log

****

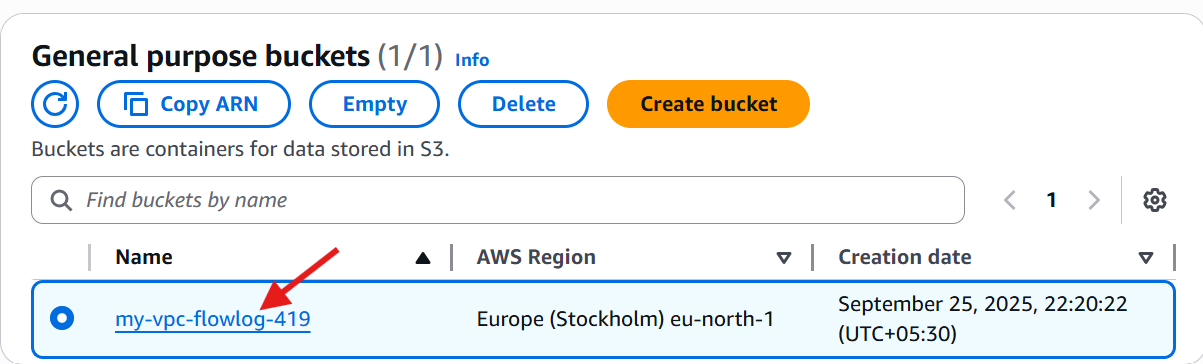
****

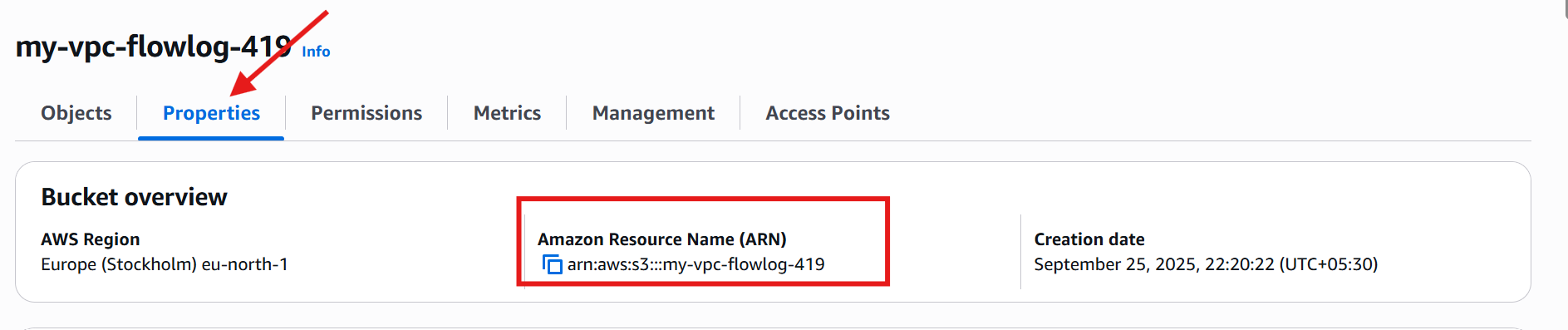
* Give maximum aggregation interval 1 minute
* Destination 🡪 send to an amazon s3 bucket
* Create Amazon resource number ARN
* Give bucket name
* And click on Create bucket

****

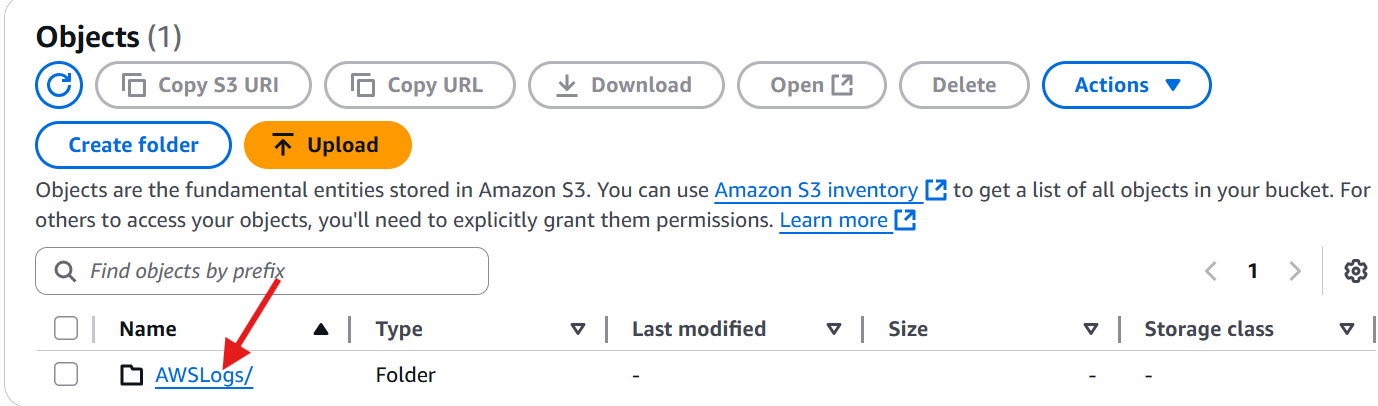
****

* Click on my vpc flowlog
* Click on properties , it shows **ARN** name
* Copy and paste it in vpc and create flow logs.

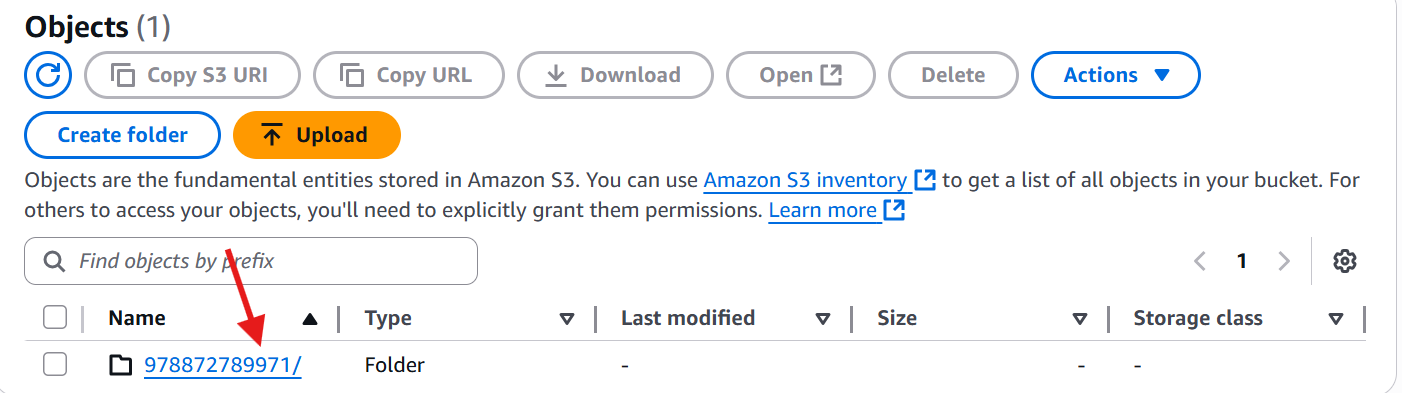
****

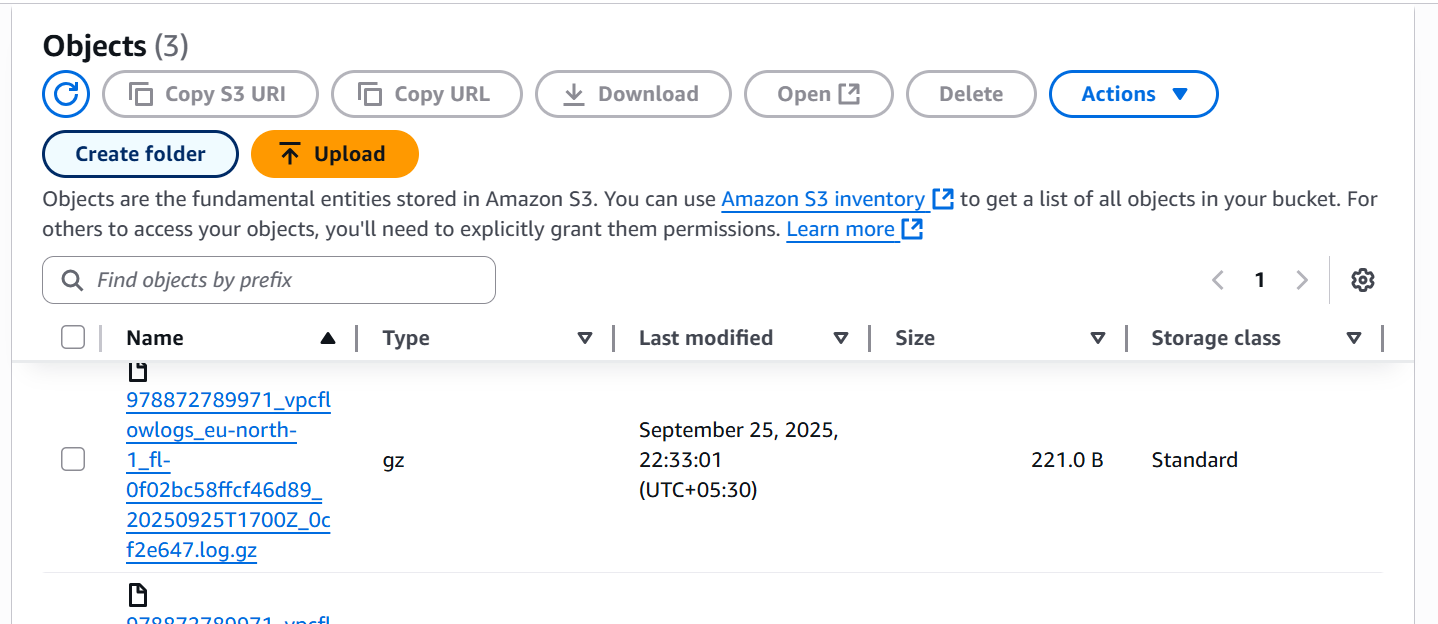
****

* **Go to s3** and click on bucket
* click on **awslogs**

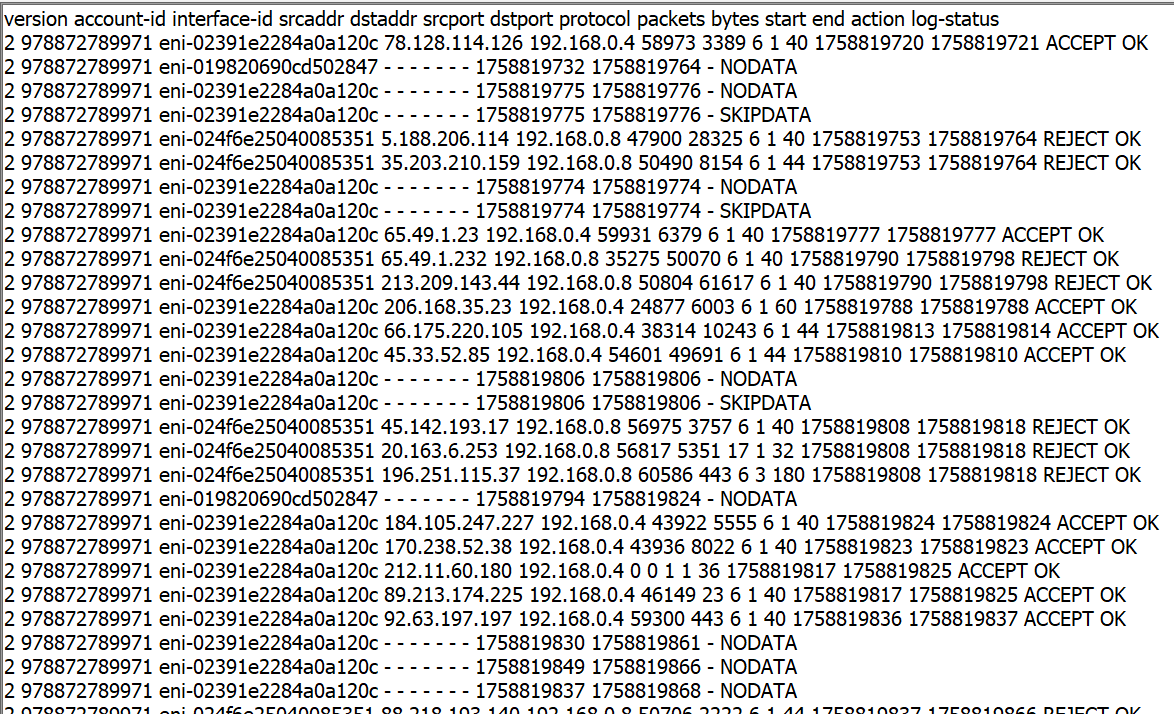
****

* **bucket-name/AWSLogs/<account-id>/vpcflowlogs/<region>/<year>/<month>/<day>/**

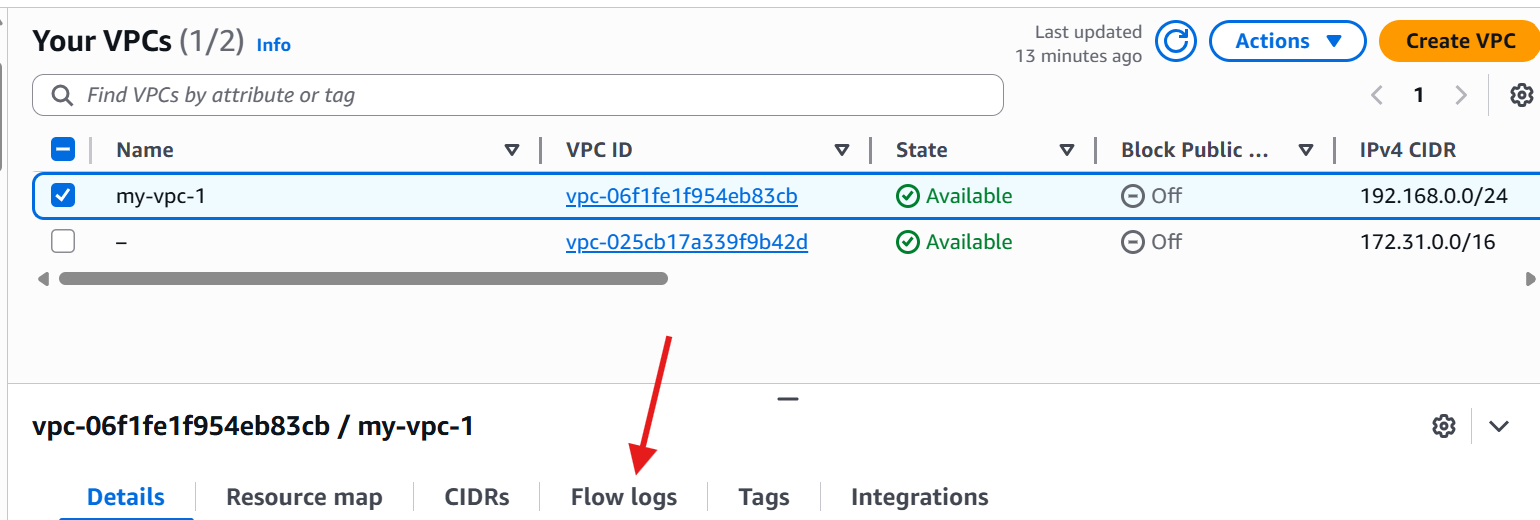
****

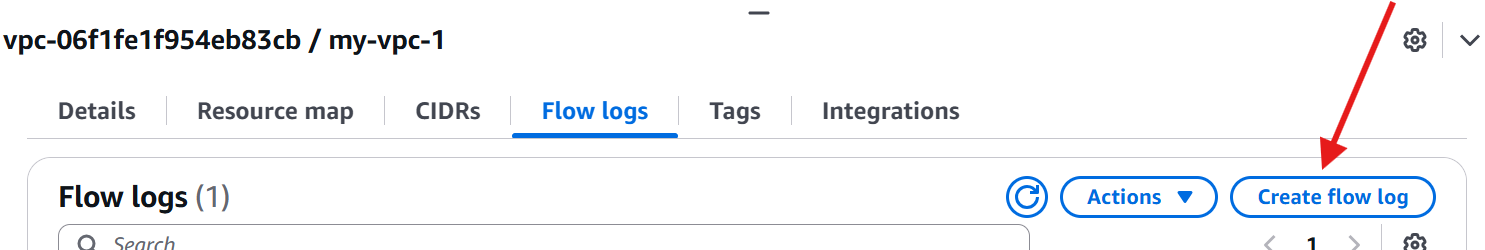
****

* Download and see the flow logs

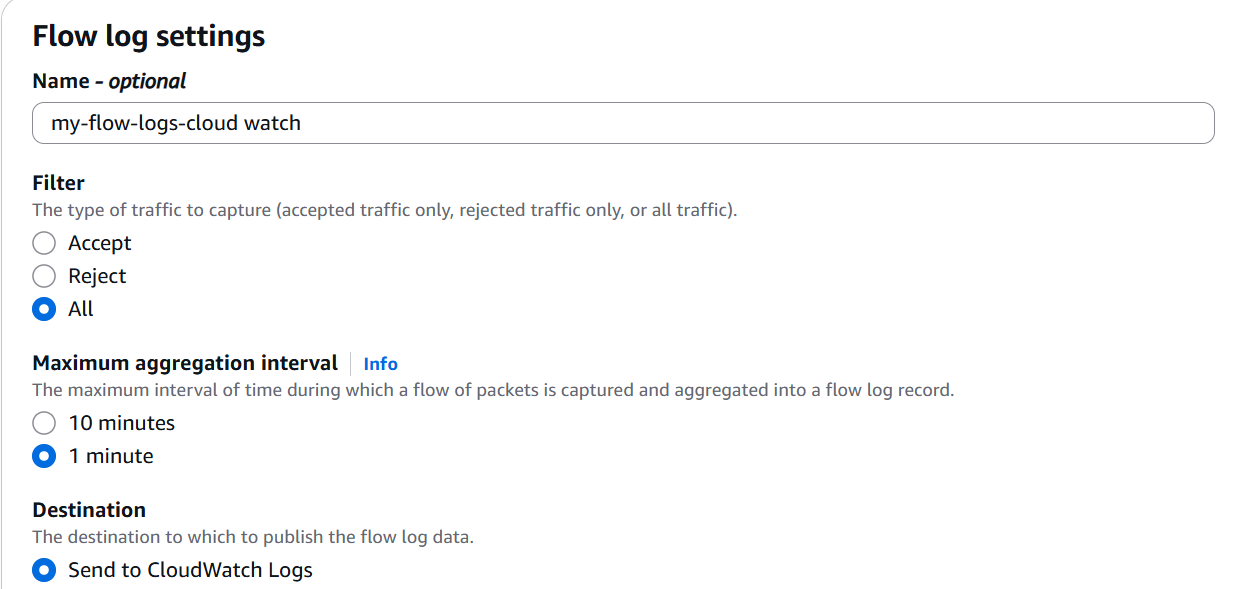
****

* **Cloud Watch flow logs**
* Go to vpc and click on your vpc
  + - Click on create flow log

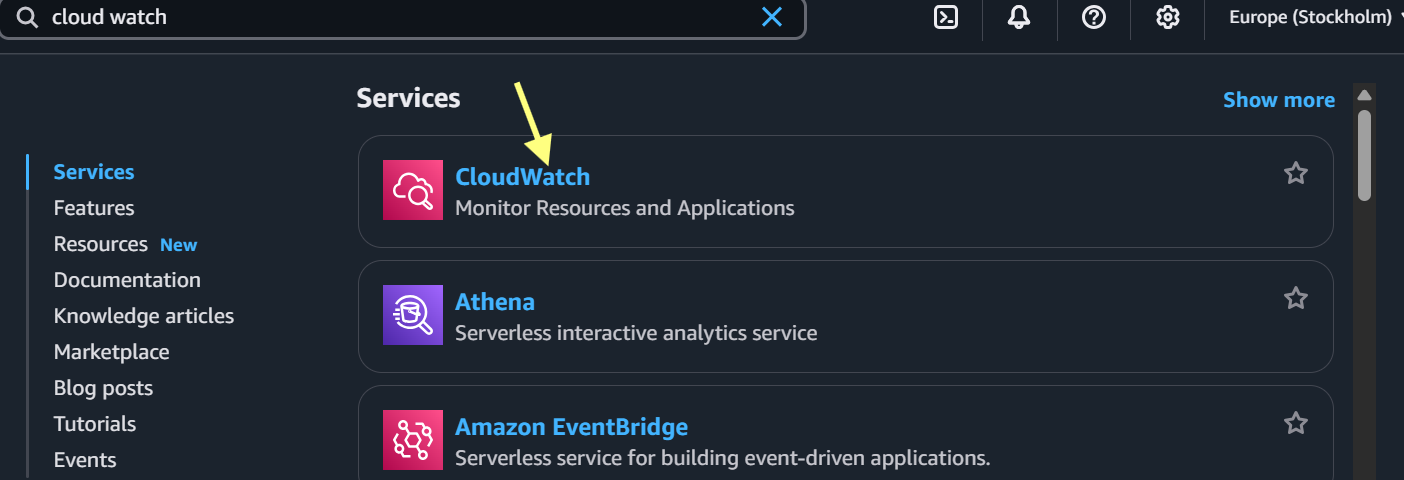
****

****

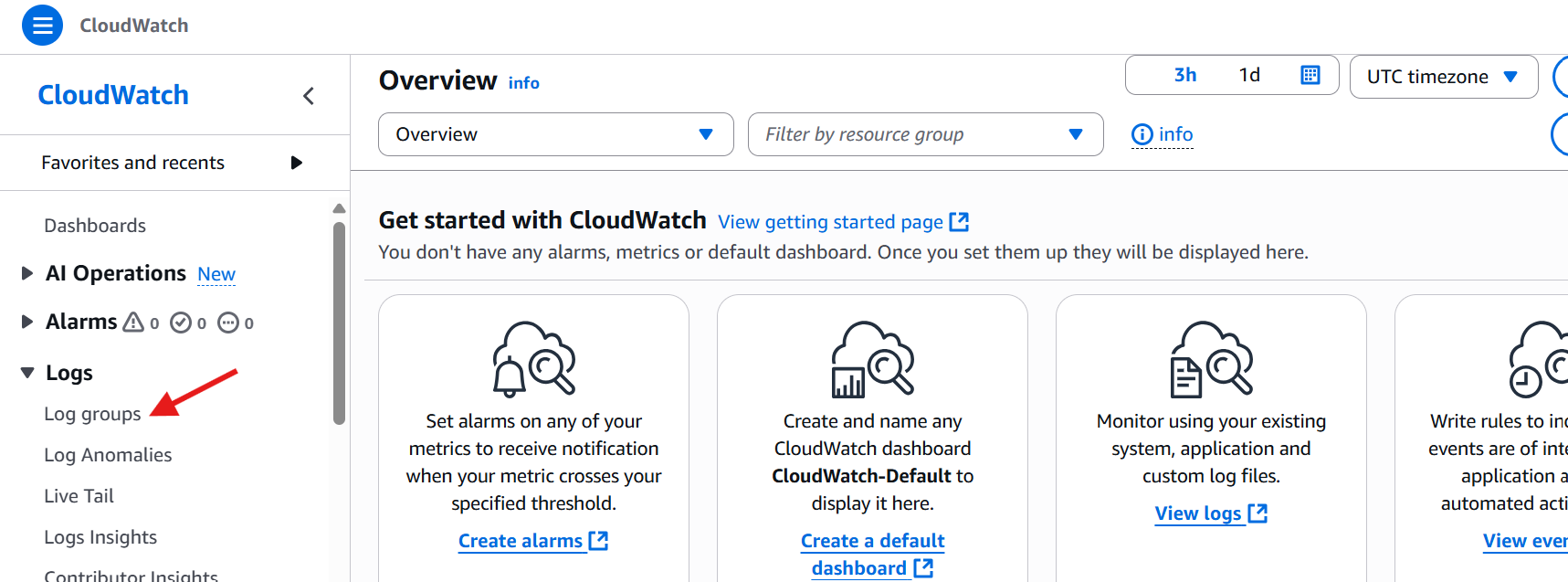
* Give name like **my flow log cloud watch**
  + - Give filter **All**
    - Give maximum aggregation interval **1 minute**
    - **Destination 🡪** Send to cloudwatch logs

****

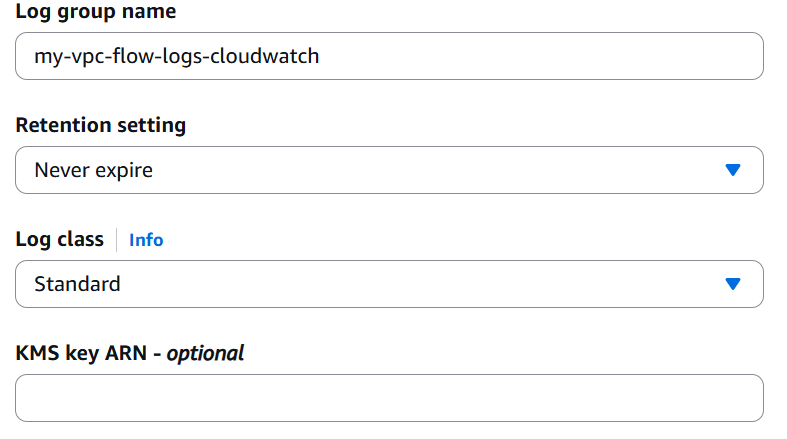
* After that it asking like **Destination log group**
* **Now,** go to ec2 and search cloud watch
* click on cloud watch

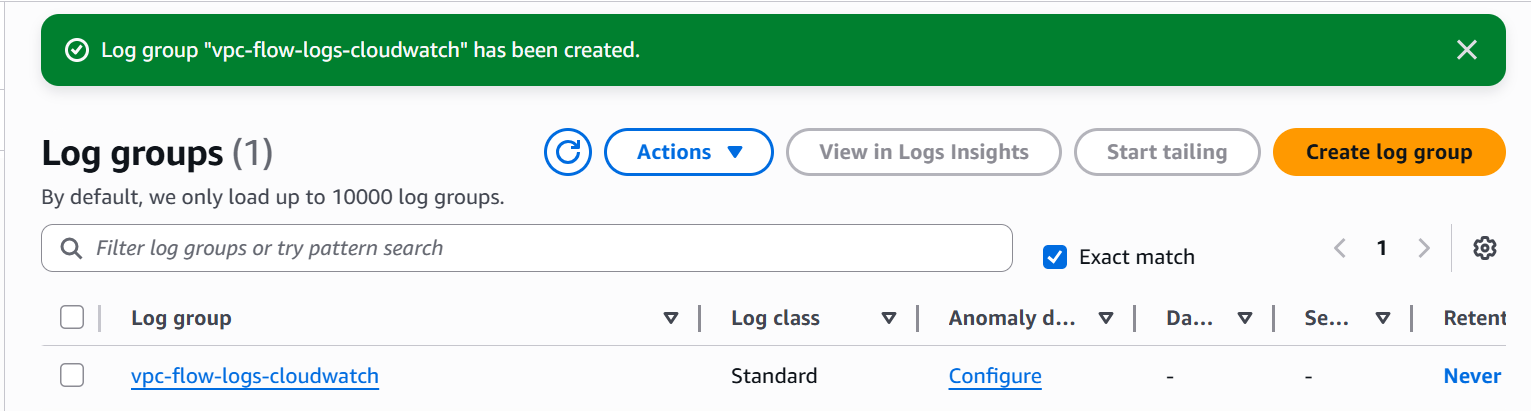
****

* In left menu , click on **Log groups**



* Click on log groups
* Click on **Create log group**
* Give lo**g group name**
* And click on **create**



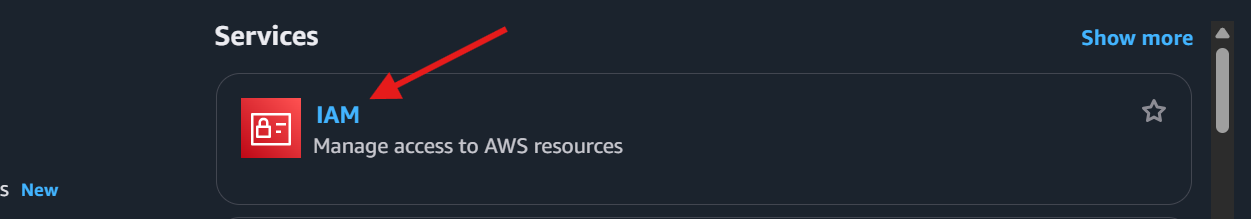


* Click on log group file
* It shows log details and there we have AWS **ARN**
* Copy that ARN and paste it in vpc flow log creation



**Then create the IAM role with policy for vpc flow log to store logs in   
cloud Watch:  
IAM Policy:**

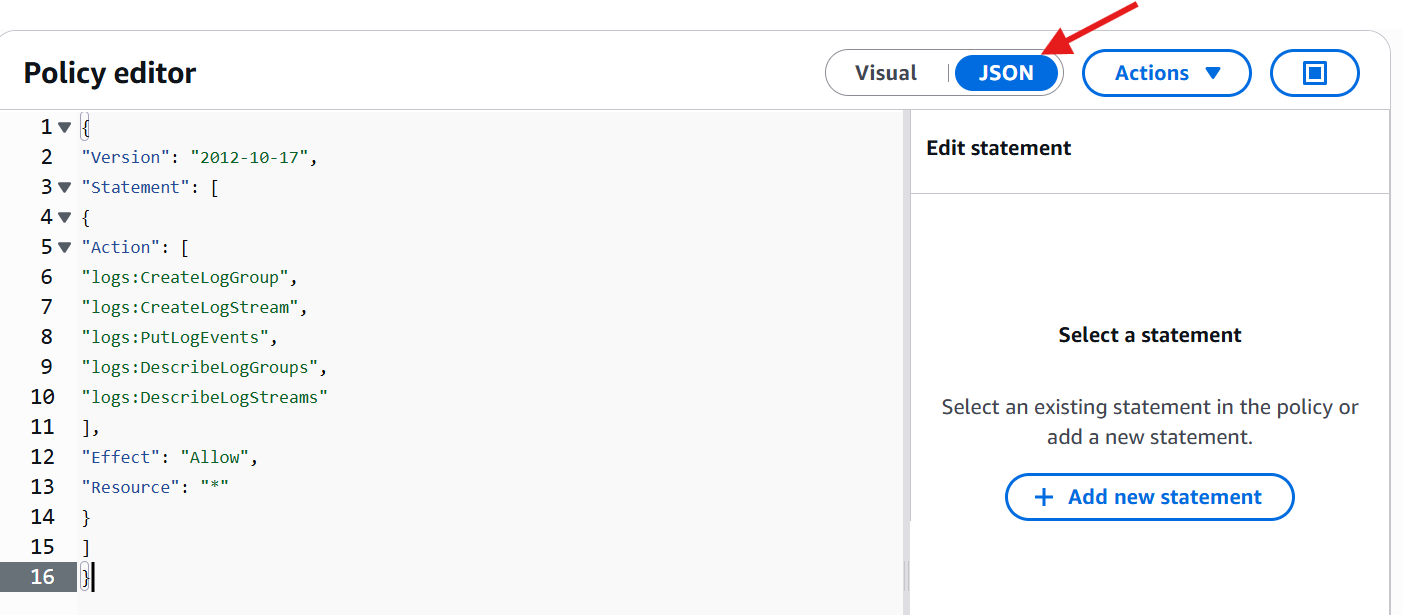
* Go to IAM



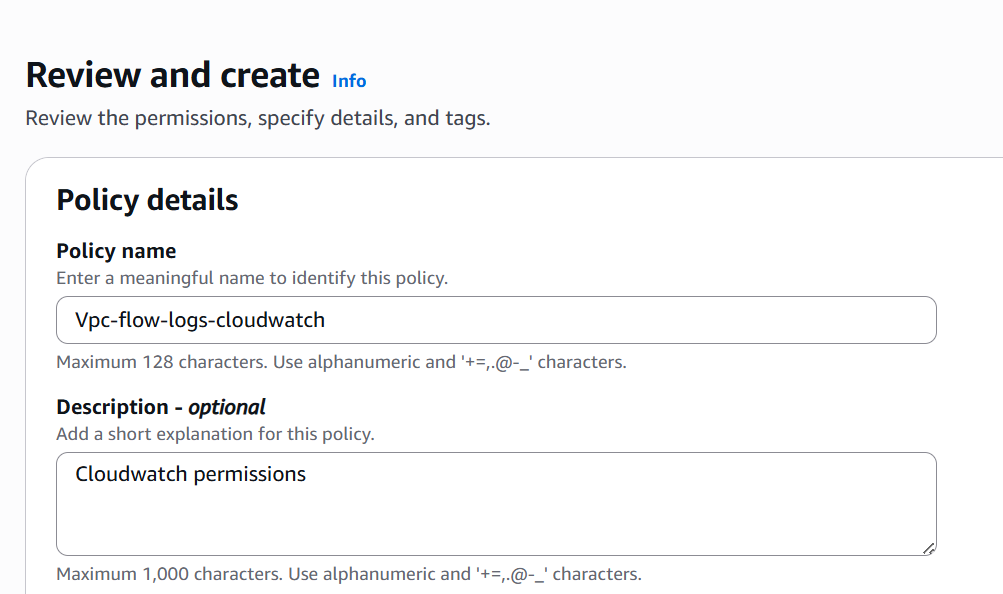
* Click on **policies** and click **create policy**



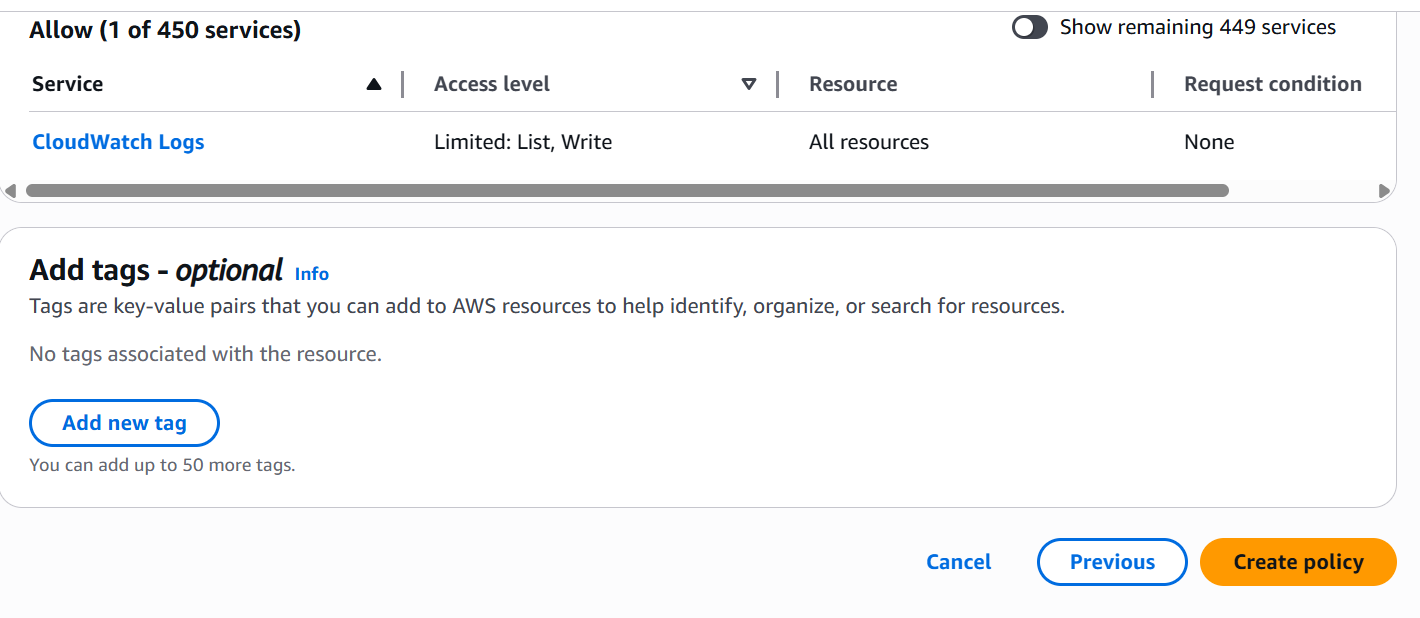
* Select Json and write the code
* Click next

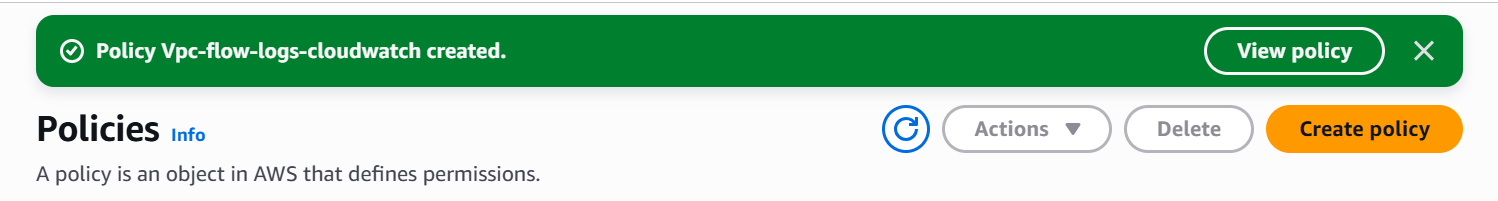


* Create policy name and give description

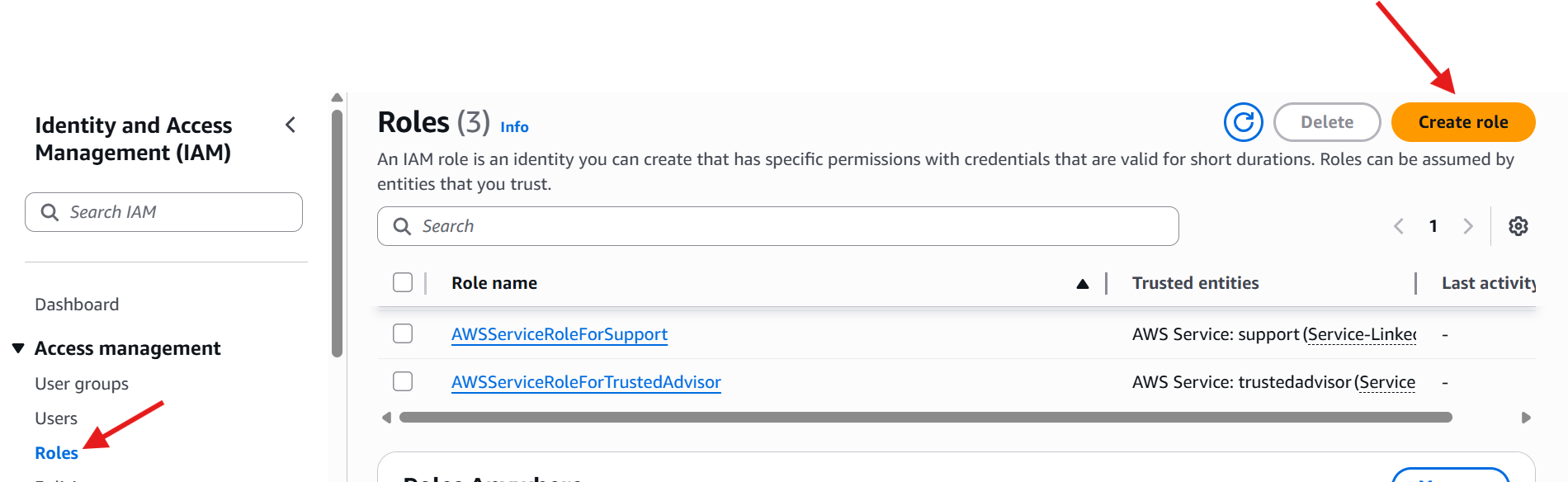


* Click on create policy

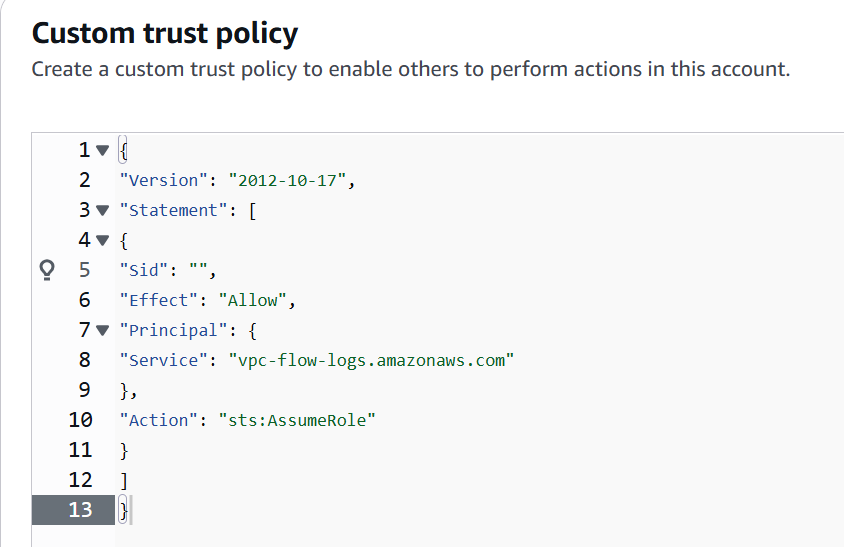




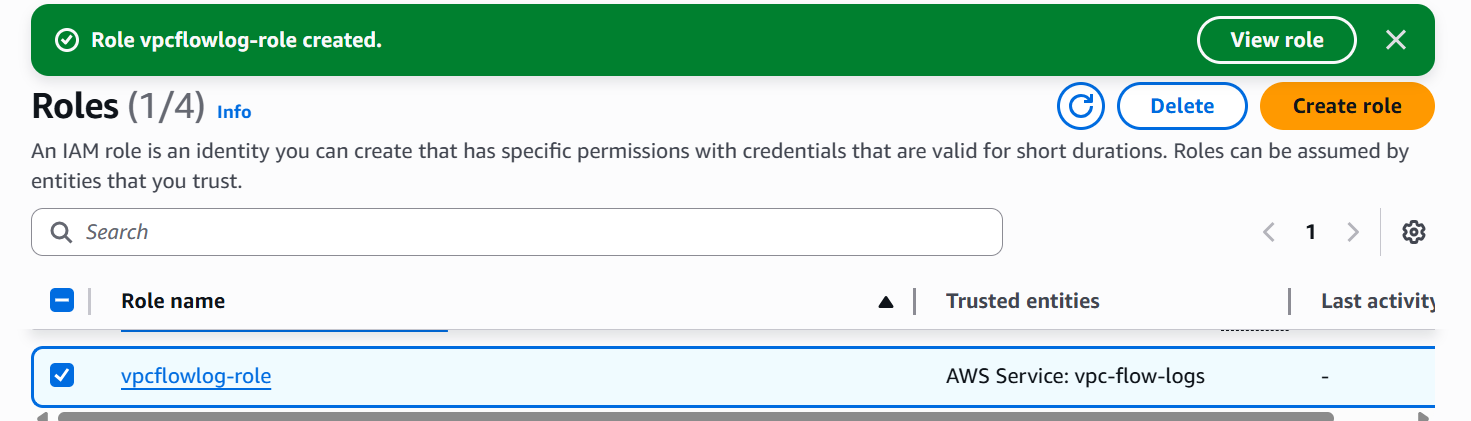
* With the help of policy create one role
* Click on roles and click create role



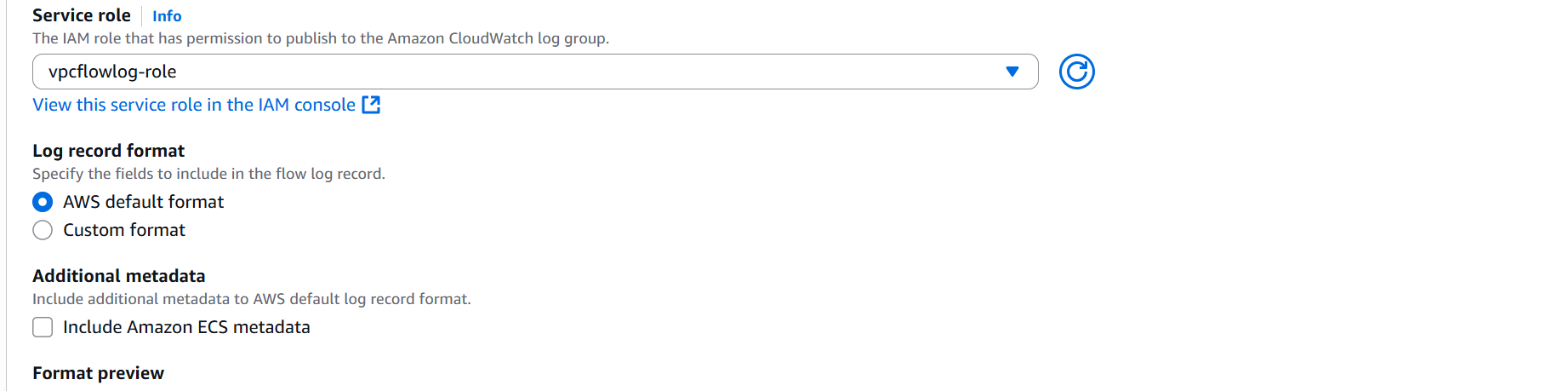
* Select trust entity type - custom trust policy
* Write the code
* Click next

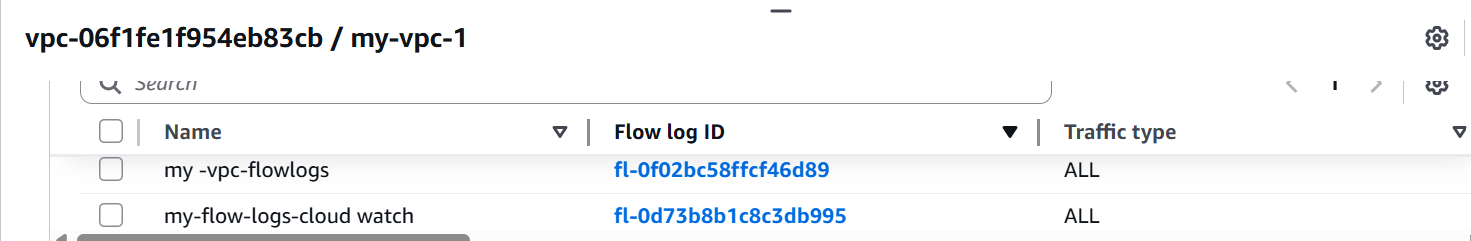


* Click on created policy and click next
* Give role name and click on create role

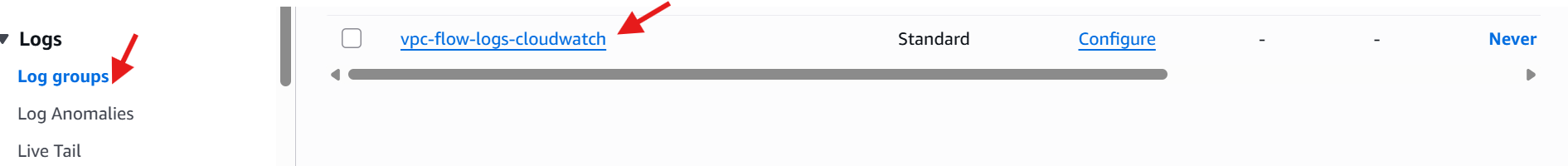


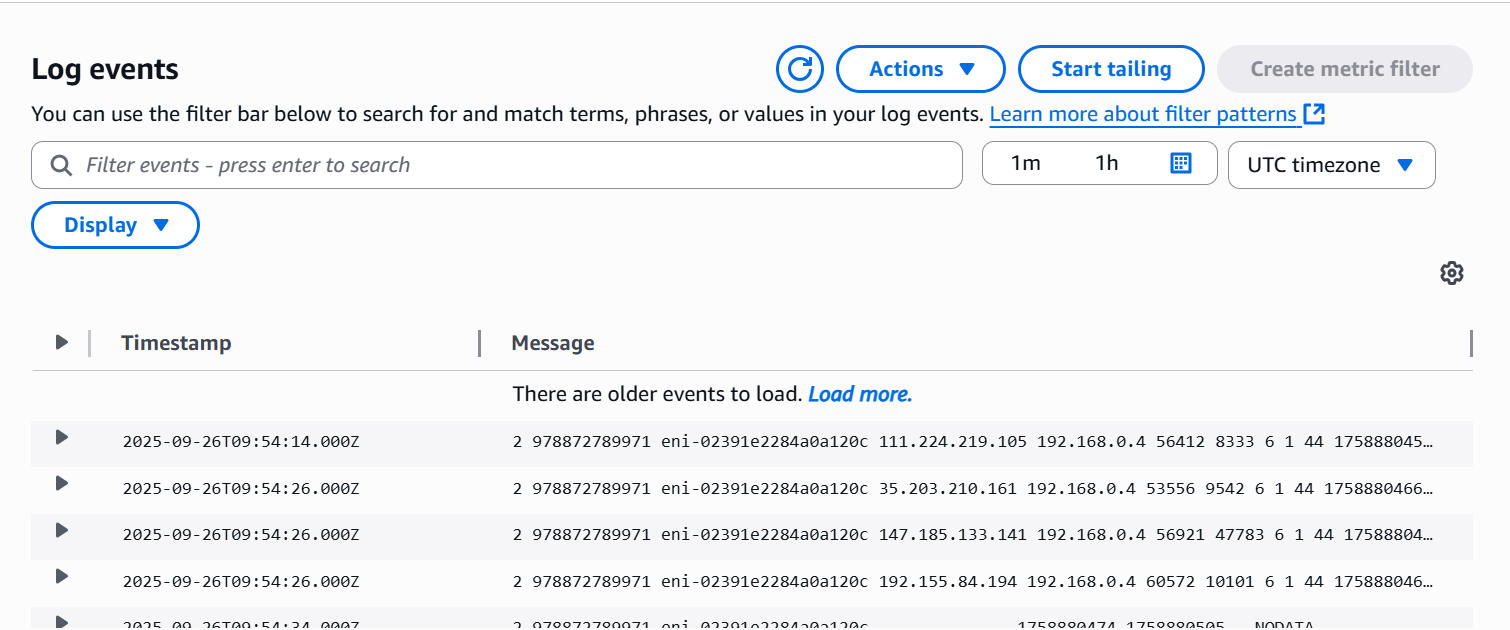
* Go to vpc and give IAM role
* Click on create flow log





* Go to cloud watch
* Click on log groups
* Click log group file and you got one log stream file
* Click on log stream file you got the cloud flow logs





-------------------------------------------------------------------------------------------------------------