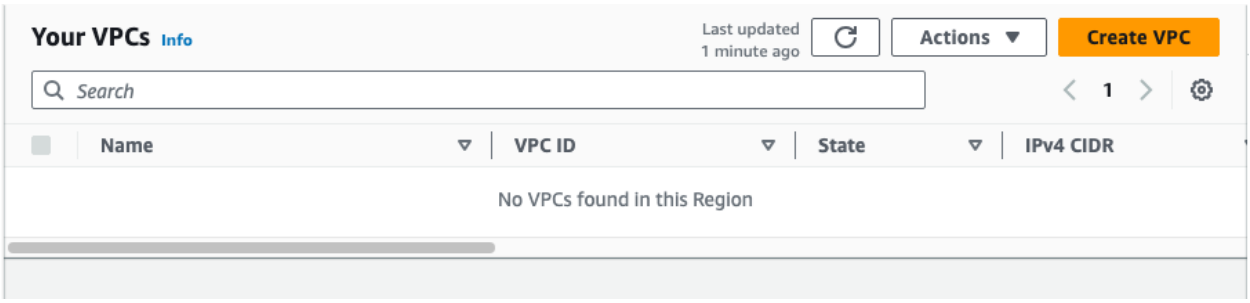
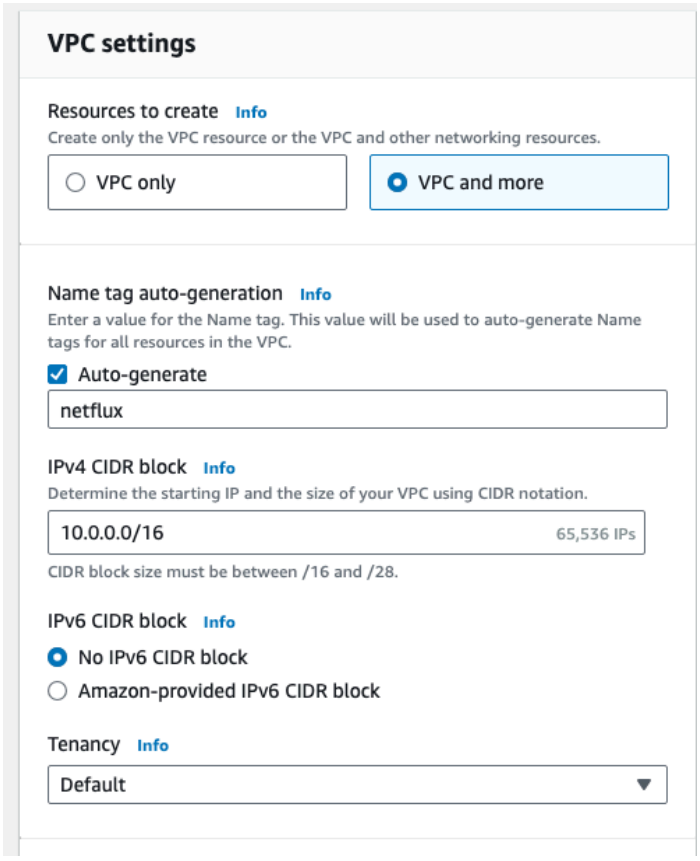
Virtual Private Cloud

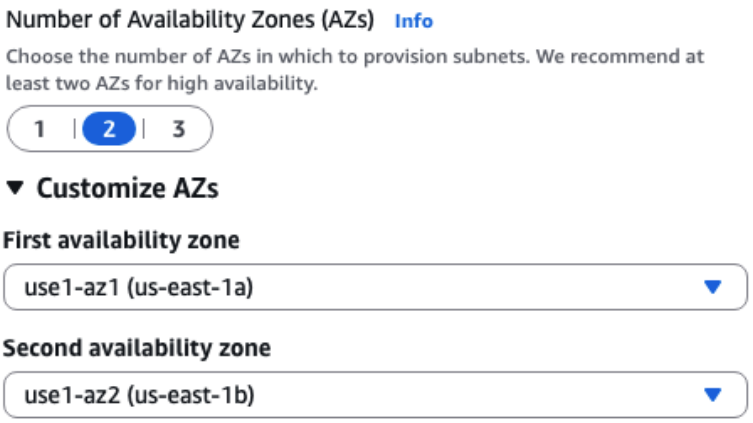
- Let's start from scratch. Go to the VPC section.
- Delete the default VPC to avoid confusion.
    - No worries, you can always create the default VPC when you need it!



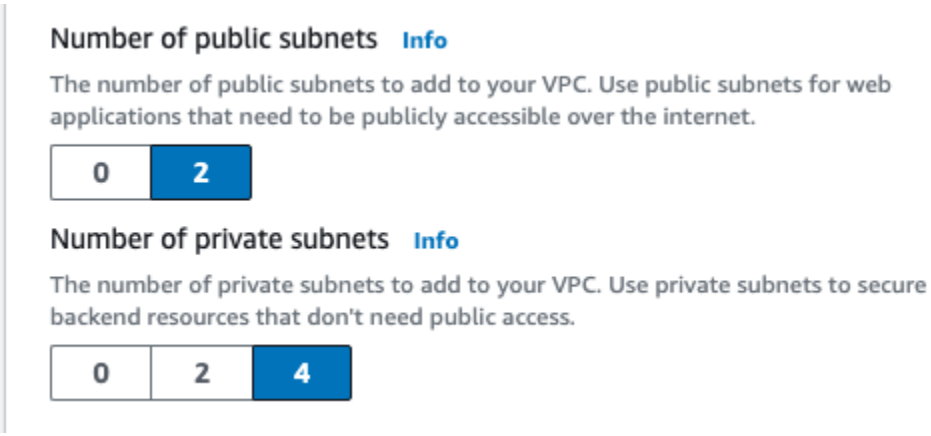
- VPC Settings
    - I give the VPC name as “netflux”.

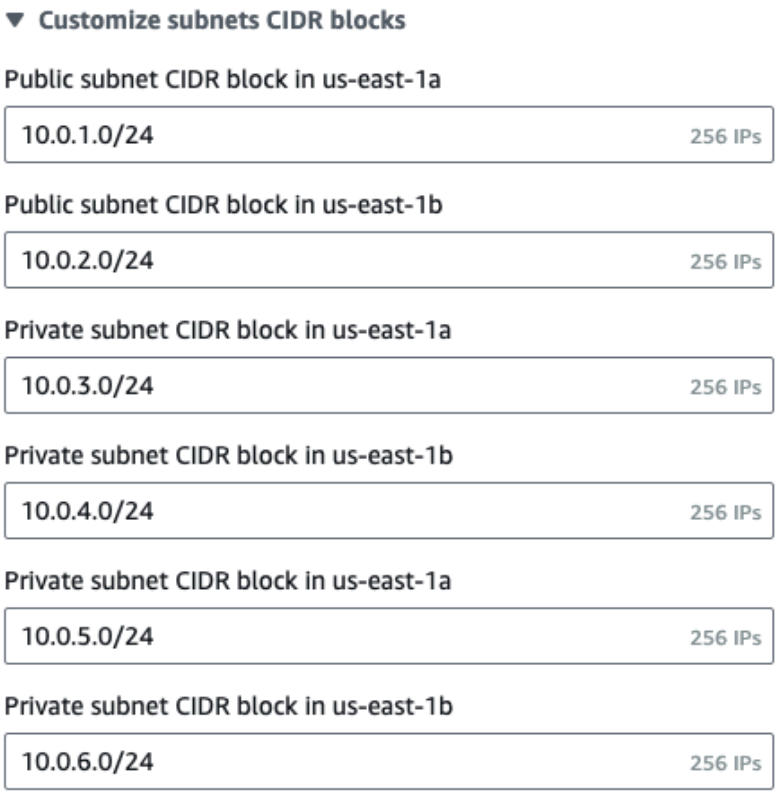


- For high availability, let’s use at least 2 AZs.

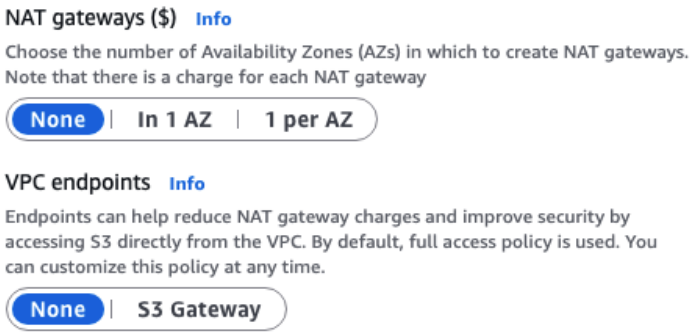


- We need
  - 2 public subnets. 1 for each AZ.
  - 2 private subnets for our backend microservices. 1 for each AZ.
  - 2 DB subnets.

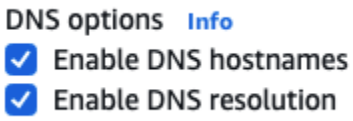




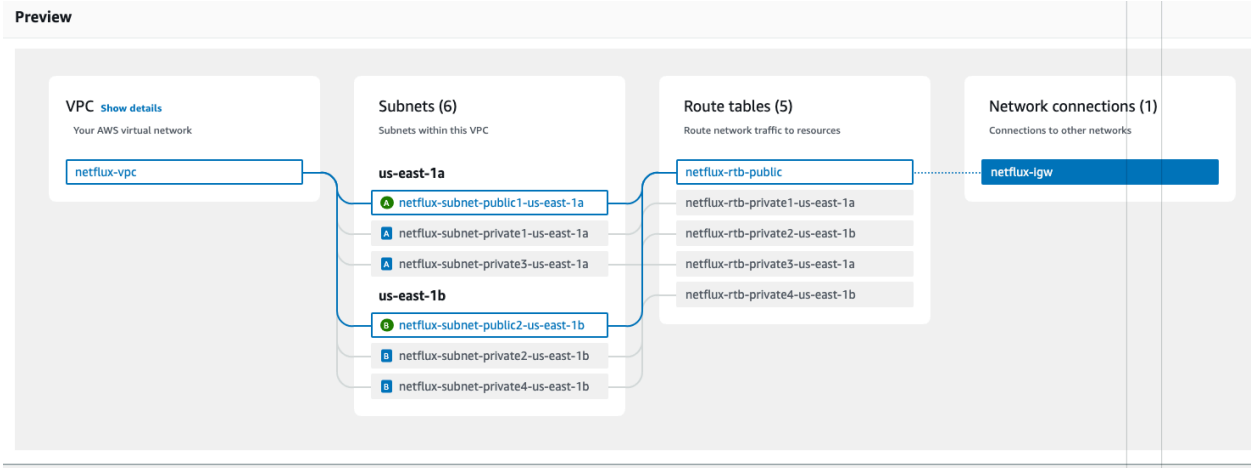
- We also need a NAT gateway. **It costs money**. We can do it later when the time comes! Let's ignore it for now. We also do NOT need ***S3 Gateway***,



- Ensure that these DNS options are selected.



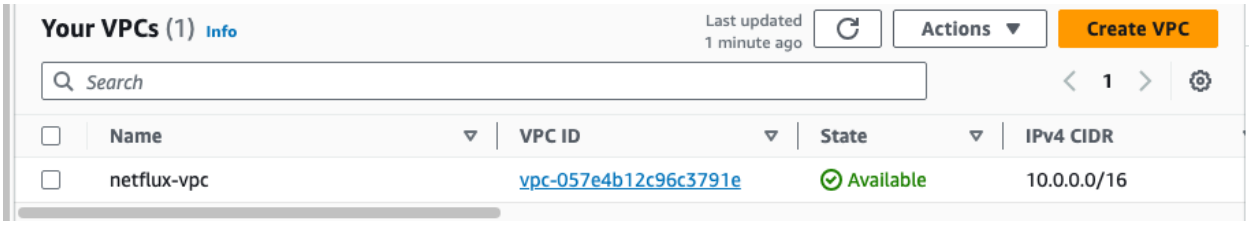
- The network we create would look like this.



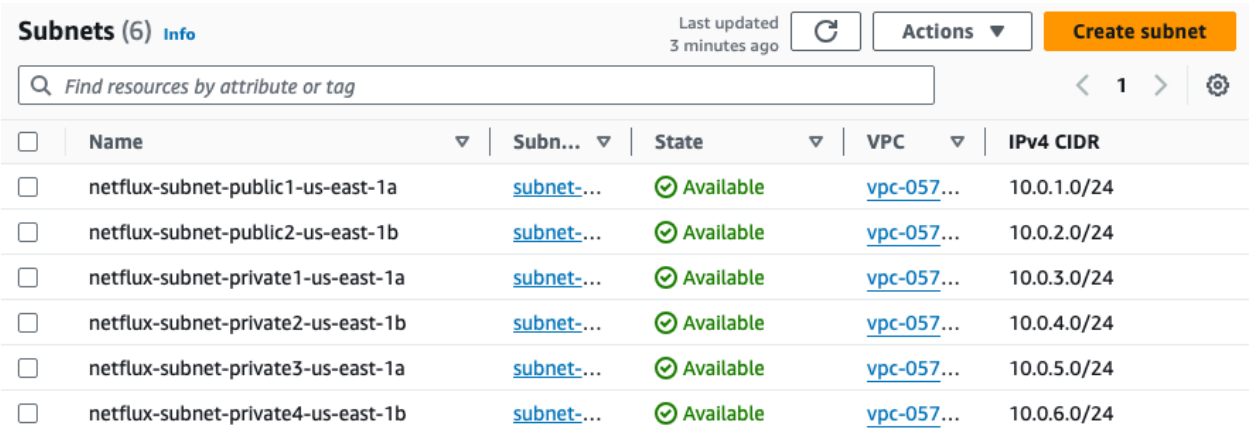
- Create VPC.
- Once it is created, do a ***Hard browser refresh*** (AWS Console refresh does NOT seem to work well in some cases).

Verify What We Have Created

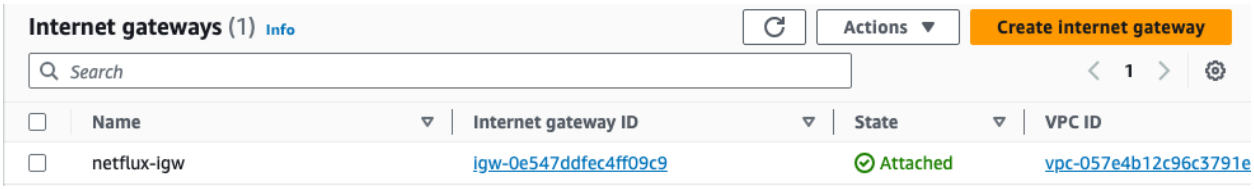
- VPC



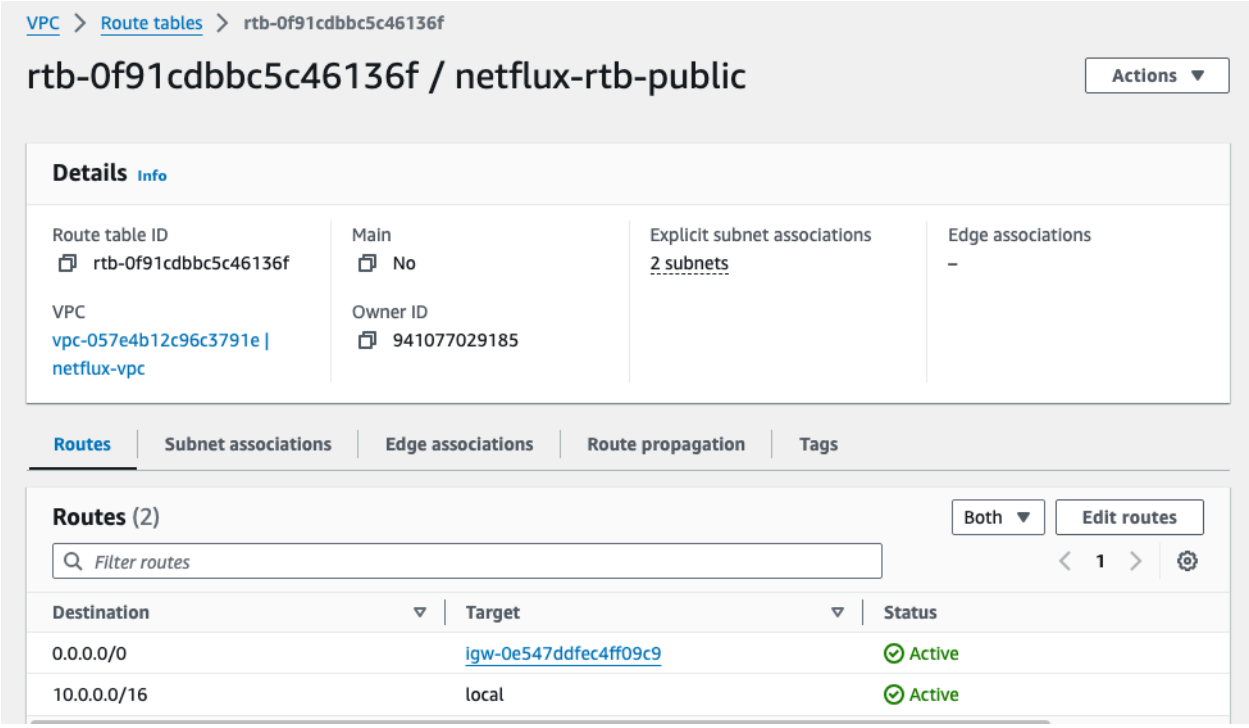
- Subnets



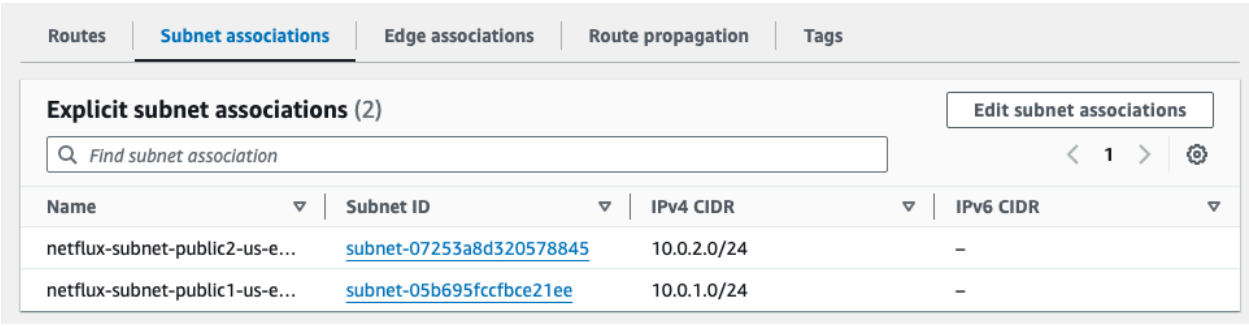
- Internet Gateway should have been attached to the VPC.



- Check the route tables. The public route table should have a route to 0.0.0.0/0 to Internet Gateway.



- Check the subnets association. 2 public subnets should have been associated with this route table!



- **Network ACL**. By default we allow all the inbound requests!

Details | Inbound rules | Outbound rules | Subnet associations | Tags

Inbound rules (2)

Edit inbound rules

Filter inbound rules

1

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	All traffic	All	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny