

ADD VPC Service to AWS :

- Open AWS → search VPC → Create VPC → Name: ditisslab → autogenerate: tick → ipv4 →

The screenshot shows the AWS Management Console interface for creating a new VPC. The breadcrumb navigation at the top indicates the path: VPC > Your VPCs > Create VPC. The main heading is 'Create VPC' with an 'Info' link. Below the heading is a descriptive paragraph about VPCs. The 'VPC settings' section contains three main configuration areas: 'Resources to create' with radio buttons for 'VPC only' and 'VPC and more' (the latter is selected and underlined in red); 'Name tag auto-generation' with a checked 'Auto-generate' checkbox and a text input field containing 'ditisslab' (underlined in red); and 'IPv4 CIDR block' with a text input field containing '172.20.0.0/16' (underlined in red) and a display of '65,536 IPs'. To the right, a 'Preview' section shows the VPC name 'ditisslab-vpc' and a 'Show details' link.

No. of availability zones : 1 → NAT Gateways : (In 1 AZ) → DNS : Tick both boxes → **Create VPC .**

- Will install → Click **View VPC.**

Create EC2 Instances :

Private Instance :

- **Launch instance** → (same as creating EC2 instances) →
- **Give name(private)** → OS: Debian → Create new key → name : (pem) → select **.pem** → Allow ALL traffic : HTTP & HTTPS →
- **Network settings** → VPC required: → select (VPC name eg. ditisslab) →

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-0a9f72fb2cf5a1efe (ditisslab-vpc)
172.20.0.0/16



- Subnet → Select: (ditisslab-subnet-private) →

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-0a9f72fb2cf5a1efe (ditisslab-vpc)
172.20.0.0/16



Subnet [Info](#)

subnet-02cb9f6345bcd734a (ditisslab-subnet-private1-us-east-1a)
VPC: vpc-0a9f72fb2cf5a1efe Owner: 092117194064 Availability Zone: us-east-1a
IP addresses available: 4091 CIDR: 172.20.128.0/20



[Create new subnet](#)

- → **Launch instance .**

Public Instance :

- EC2→Launch instances → Name: (Public) → Os : debian →Create key pair → Name: public→ select : .ppk → Not allow http,https→
- Network settings: Select: ditisslab-vpc→ select: public subnet → Auto-Assign public IP : Enable → **Launch instance .**

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-0a9f72fb2cf5a1efe (ditisslab-vpc)
172.20.0.0/16



Subnet [Info](#)

subnet-050aad31f3586c37b (ditisslab-subnet-public1-us-east-1a)
VPC: vpc-0a9f72fb2cf5a1efe Owner: 092117194064
Availability Zone: us-east-1a IP addresses available: 4090
CIDR: 172.20.0.0/20



[Create new subnet](#)

Auto-assign public IP [Info](#)

Enable

**Connect public instance machine to putty → Go to putty → Public ip → public key
→ save session → Login**

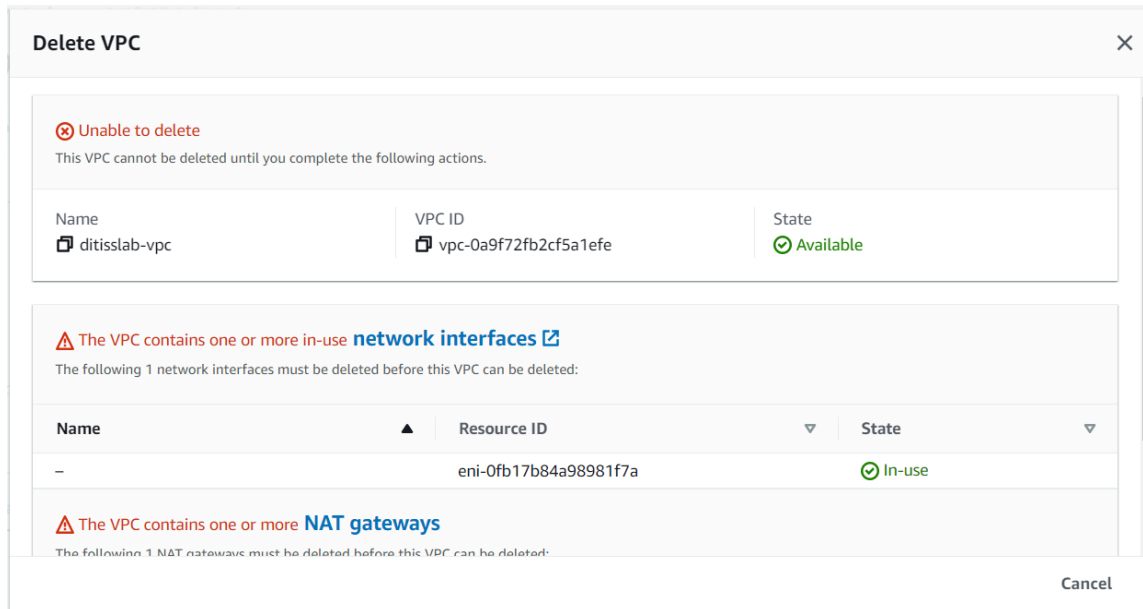
- **Commands :**

- nano key
 - *paste private key from private.pem file*
- ssh -i key admin@private ip(private instance) → (will give error)
- chmod 400 key
- ssh -i key admin@private ip(private instance)
- We will give in private machine
- sudo apt-get update
- sudo apt install apache2
- sudo rm /var/www/html/index.html
- sudo nano /var/www/html/index.html
 - Type : (Private server)
- Exit → We will go in public machine
- curl (private ip) → we will see private machine browser
- links (private ip) → we will see private machine browser

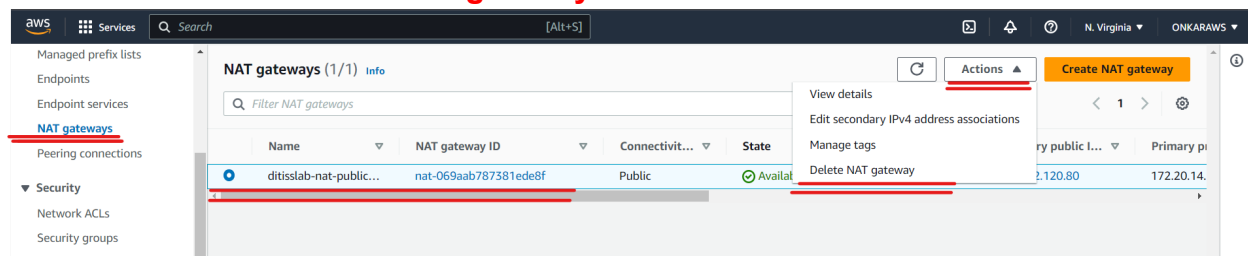
IMP :

Remove VPC : →

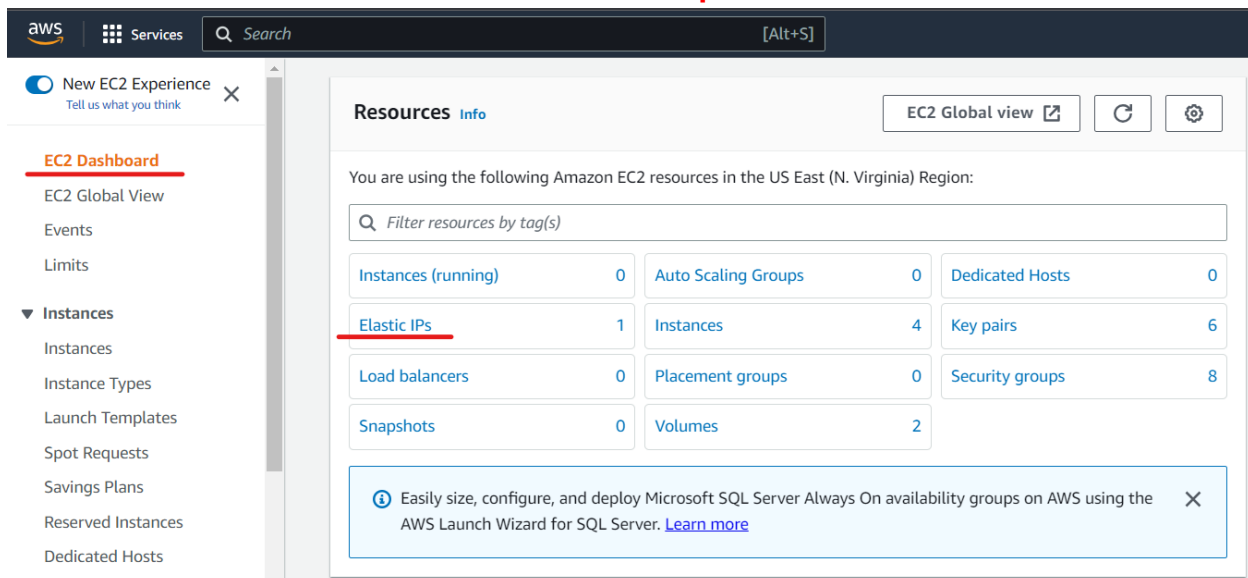
- **Terminate both instances→**
- **Then Go to All VPC → delete VPC created .**
 - **Will show 3 error red marks : →**



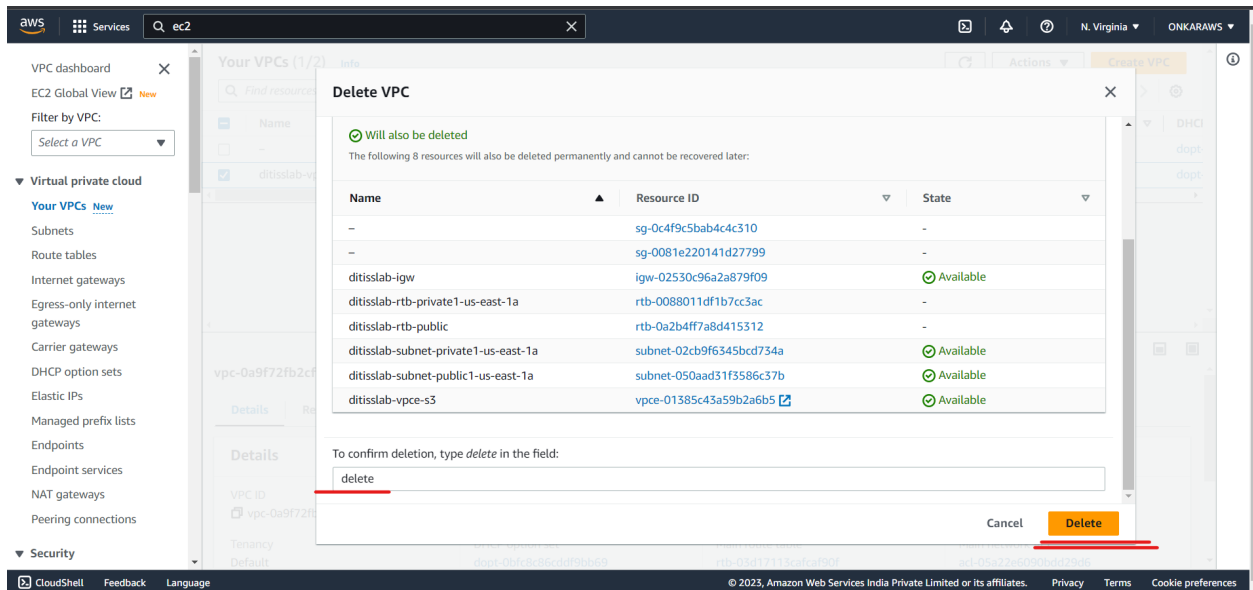
1. Remove NAT gateway: → At left side → Actions → Delete NAT gateway



2. → EC2 Dashboard → Elastic IP's → Select all → Actions : → Release elastic ip address



3. Then the VPC finally will delete : DELETE



IMP: There should be Nothing in “ Your VPC’s ” →

- 1. Elastic IP’s**
- 2. NAT gateways**

Total no. of active regions should be : 0