

# VPC

Login to aws

VPN

Delete default vpn

Check vpns, subnet, routing table

Create VPN

Name: snuc-vpn

IPv4 CIDR : 192.168.0.0/16

Create vpn

Create subnet

Select above vpn

Name: public-subnet

Availability zone: mumbai

IPv4 subnet CIDR block: 192.168.1.0/24

Create subnet

Create internet gateway

Click the created igw, actions-> attach it to a vpc

Routing table

Click the created routing table

Routes-> edit routes

Add route

0.0.0.0/0 -> internet gateway-> select the created igw -> save changes

GET Two elastic ips-> click on elastic ips in the left side bar

Allocate two elastic ips - web server and web client

## Creating EC2 instance

Search bar-> ec2 -> open in new tab

Left side bar-> instances -> launch an instance

Right side-> make it 2 instances

Name: VMs

Ubuntu system

Process without key pair

Launch an instance

Attach it to public ip address:

Go to ec2-> instances

Rename the vms to web server and web client

Go to the elastic ip addresses. Associate each one with web server and web client respectively

Connect to the instance via ec2:

Go to instance- web server-> connect -> connect to ec2 instance  
IN THE WEBSERVER VM: (we need to install apache)  
ping 8.8.8.8 (checking)  
sudo apt update  
sudo apt install apache2  
service apache2 status

Go to instance- web client-> connect -> connect to ec2 instance  
IN THE WEB CLIENT VM: (we need to install links)  
ping 8.8.8.8 (checking)  
sudo apt update  
sudo apt install links

### **In security group, add http access:**

Go to web server instance-> click on the instance-> go to security-> click security group->  
Edit inbound rules-> add rule-> http, anywhere ipv4-> save rules

### **Allow ssh and http on nacl:**

NACL is on VPC service  
Go to vpc console  
Left bar-> nacls  
Click on the nacl corresponding to our vpc (snuc-vpc)  
Go to inbound rules and edit  
Remove the rules that exist  
Add 100, all traffic, allow

### **Test the web access through web client thru links:**

Go to web server vm, copy the public ip address and then got to web client vm-> links http://<ip>  
APACHE - IT WORKS!

### **~~Commands to generate the public and private key:~~**

~~mkdir key-pair-labs  
cd key-pair-labs  
openssl genrsa -out snu-privatekey.pem 2048  
openssl rsa -in snu-privatekey.pem -pubout -out snu-publickey.pem  
chmod 400 snu-privatekey.pem  
cat snu-publickey.pem~~

~~Copy paste the public key to aws console~~

## Testing ROUTE 53 service

Create a hosted zone in aws route 53 service with 11 digit registration number for ngaws.xyz  
- Search bar: route 53-> dashboard (left side bar)-> create hosted zone

- 21011101125.ngaws.xyz in domain name ->create

Login to godaddy

Url: <https://www.godaddy.com>

Username: aws-ng

Password: Welcome1!

**Get the name server(any one) information from route 53 dashboard and update NS record in GoDaddy portal for your subdomain**

In hosted zones-> go to the one you hosted just now-> copy one of the value/route traffic to

In godaddy-> go to domain-> domain settings-> DNS-> add new record->

Choose NS, 21011101125(name), paste the ns value-> save

**Create a record in your hosted zone with following details:**

Subdomain:www

Value : IP address of web server

Routing policy: simple

(hosted zones-> create record ...-> create all records)

**Check website reachability with url:**

(Go to web client)

links **www.<reg no>.ngaws.xyz**

**Nslookup: shows you the ip address of the non authoritative access**

# IAM-identity and access management

Search bar-> iam

users-> create user-> alice, provide access-> i want iam user-> next->next->create user

Save the credentials:

<https://070781912515.signin.aws.amazon.com/console>

alice

x89acG0]

Sign in to alice-> change password-> and you can see that all permissions are denied

Go back to users

Click alice-> add permissions-> attach policies directly-> search for ec2-> give ec2fullaccess permissions policy alone.

Now refresh and you can see in alice login that instances can be created

To apply the same policy to many users, we can create a group. iam-> user group-> create group-> add users -> give the ec2fullaccess policy-> create user group

Now you can create bob user and give vpc access. When u check you will be able to create ec2 instances also though as vpc access allows some ec2 access. Bob cant access route 53 services such as creating a hosted zone

Side Note: You can create account alias