VPC

Trainee Name : Chhavi Sharma
Newers ID : 4023
Mentor Name : Nishith Kulshrestha
College : UPES

1. When to use Elastic IP over Public IP

Ans.

Public **IP addresses** are dynamic - i.e. if you stop/start your instance you get reassigned a new public IP. Elastic **IPs** get allocated to your account, and stay the same - it's up to you to attach them to an instance or not. You could say they are static public **IP addresses**.Elastic IP addresses also have the advantage of being dynamically re-mappable. So, rather than being assigned to a particular EC2 instance for the life of the instance, they can be assigned to different EC2 instances as necessary - for example to switch between instances while an upgrade is being performed, or to direct traffic to a second instance if the first instance fails.

2. Valid IP Ranges for LAN, Implication of using Public IP ranges for Private Network.

Ans.

1. Class A : 10.0. 0.0 – 10.255. 255.255.
2. Class B : 172.16. 0.0 – 172.31. 255.255.
3. Class C : 192.168. 0.0 – 192.168. 255.255.

We can use a public IP-Address-Range in our private network. There is no law against this. But we have to take precautions to avoid any routing-trouble when a machine with an IP-Address that actually belongs to a public range wants to access the internet.

3. List down the things to keep in mind while VPC peering.

Ans.

1.To enable flow of traffic between the VPCs using private IP addresses, the owner of each VPC in the peering connection must manually add a route to one or more of their VPC route tables that points to the IP address range of the other VPC(i.e. The peer VPC).

2.The owner of the requester VPC sends a request to the owner of the acceptor VPC to create a VPC peering connection. The acceptor VPC can be owned by you , or another AWS account, and cannot have a CIDR block that overlaps with the requestor VPCs CIDR block.

3.You have a quota on the number of active and pending VPC peering connections that you can have per VPC.

4.You cannot have more than one VPC peering connection between the same two VPCs at the same time.

4. CIDR of a VPC is 10.0.0.0/16, if the subnet mask is /20 calculate the number of subnets that could be created from the VPC. Also find the number of IP in subnet.

Ans.

Number of Subnets :

20 - 16 = 4

Therefore, 2^4 = **16 subnets**.

Now, The number of IPs in each subnet is

32-20 = 12

Therefore, 2^12 = **4096 IPs**

5. Differentiate between NACL and Security Groups.

Ans.

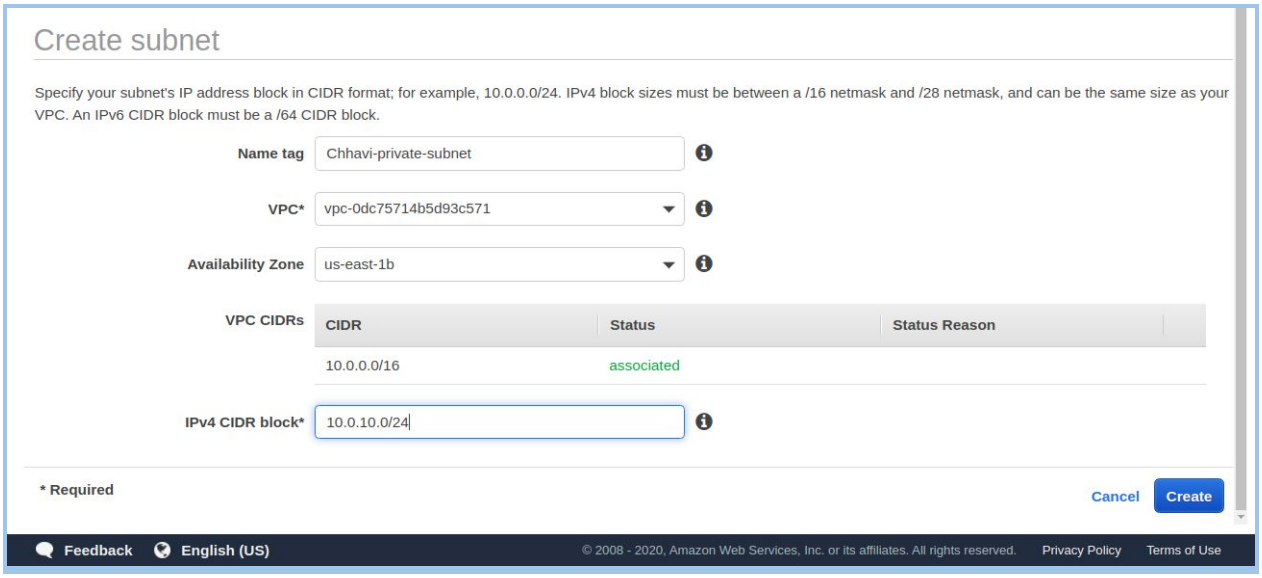
Sno.	NACL	Security Groups
1.	Stateless	Stateful
2.	Operates at subnet level	Operates at instance level
3.	Supports allow and deny rules	Supports only allow rules
4.	Rules are evaluated until matched(in the order of the rule number specified)	All the rules are evaluated when deciding whether to allow traffic.
5.	Applied to all instances within the subnet.	Applied to a particular instance during launch configuration

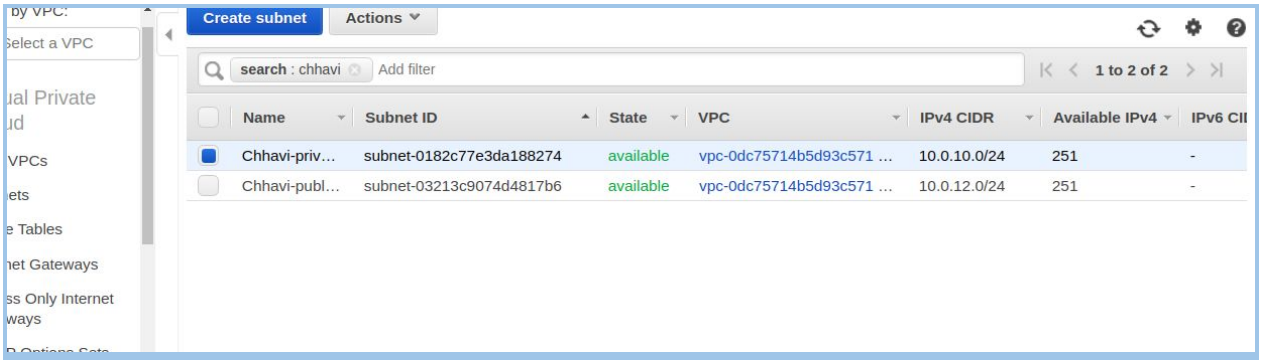
6. Implement a 2-tier vpc with following requirements:

1. Create a private subnet, attach NAT, and host an application server(Tomcat)

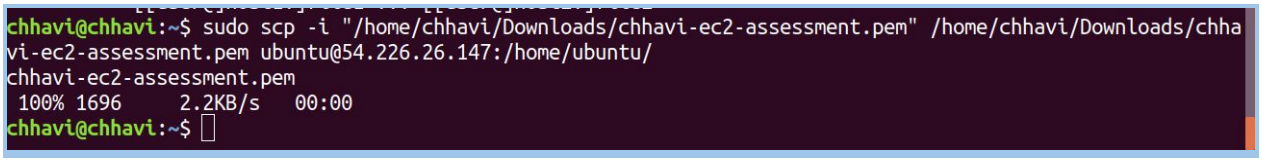
Ans.

1. Create a subnet





Copied the pem file from the local machine to public instance .



ssh into the private instance from public instance and install tomcat9

```
ubuntu@ip-10-0-12-241:~$ sudo ssh -i /home/ubuntu/chhavi-ec2-assessment.pem  ubuntu@10.0.10.214
The authenticity of host '10.0.10.214 (10.0.10.214)' can't be established.
ECDSA key fingerprint is SHA256:7yenJsY0HRv//4GtxYkWVTOb5fAuh/CxG312yEL6CiM.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.10.214' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Feb 24 14:50:18 UTC 2020

  System load:  0.0               Processes:           86
  Usage of /:   13.6% of 7.69GB   Users logged in:     0
  Memory usage: 15%               IP address for eth0: 10.0.10.214
  Swap usage:   0%

0 packages can be updated.
0 updates are security updates.
```

```
Get:26 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [218 kB]
Get:27 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [6760 B]
Get:28 http://security.ubuntu.com/ubuntu bionic-security/multiverse Translation-en [2700 B]
Fetched 18.4 MB in 4s (4780 kB/s)
Reading package lists... Done
ubuntu@ip-10-0-10-214:~$ sudo apt-get install tomcat9
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java default-jre-headless fontconfig-config fonts-dejavu-core java-common libapr1
  libasound2 libasound2-data libavahi-client3 libavahi-common-data libavahi-common3 libcups2
  libeclipse-jdt-core-java libfontconfig1 libjpeg-turbo8 libjpeg8 liblcms2-2 libnspr4 libnss3
  libpcsclite1 libtcnative-1 libtomcat9-java libxi6 libxrender1 libxtst6 openjdk-11-jre-headless
  tomcat9-common x11-common
Suggested packages:
  default-jre libasound2-plugins alsa-utils cups-common liblcms2-utils pcscd libnss-mdns
  fonts-dejavu-extra fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei
  fonts-indic tomcat9-admin tomcat9-docs tomcat9-examples tomcat9-user
The following NEW packages will be installed:
  ca-certificates-java default-jre-headless fontconfig-config fonts-dejavu-core java-common libapr1
  libasound2 libasound2-data libavahi-client3 libavahi-common-data libavahi-common3 libcups2
```

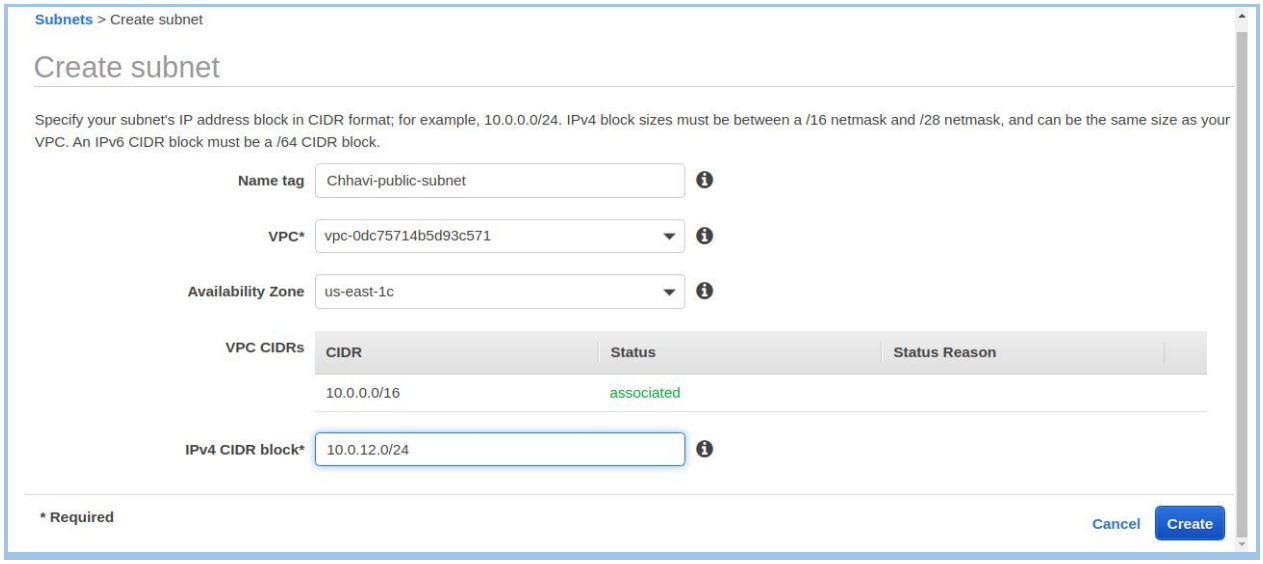
```
ubuntu@ip-10-0-10-214:/var/lib/tomcat9/conf$ curl localhost:8080
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>Apache Tomcat</title>
</head>

<body>
<h1>It works !</h1>

<p>If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations
!</p>

<p>This is the default Tomcat home page. It can be found on the local filesystem at: <code>/var/lib/tomcat9
/webapps/ROOT/index.html</code></p>

<p>Tomcat veterans might be pleased to learn that this system instance of Tomcat is installed with <code>CA
```

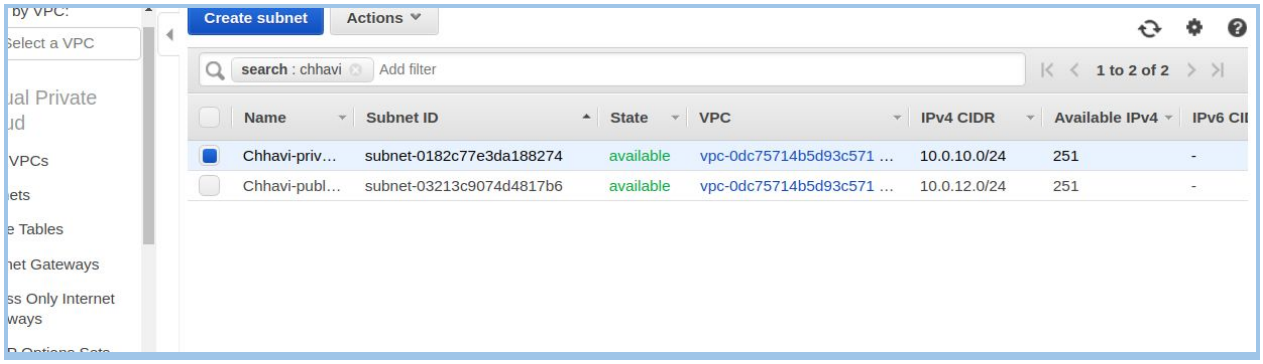
2. Create a public subnet, and host a web server(Nginx), also proxypass to Tomcat from Nginx

Ans.

Create a public subnet.

(Create a route table. Attach an internet gateway in the route table and associate it to the subnet)





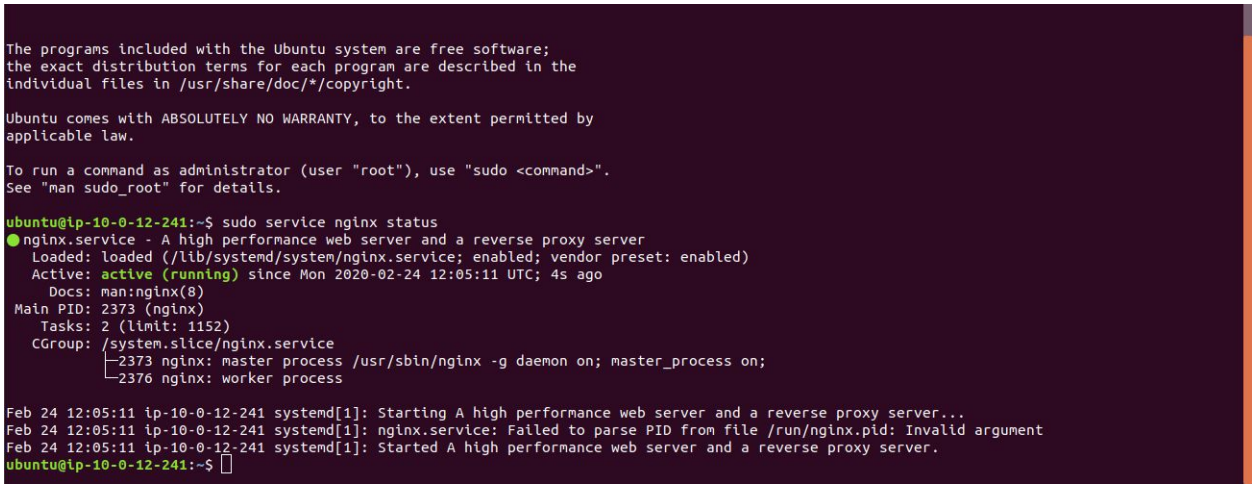
In the user data write :

#!/bin/bash

sudo apt-get update -y

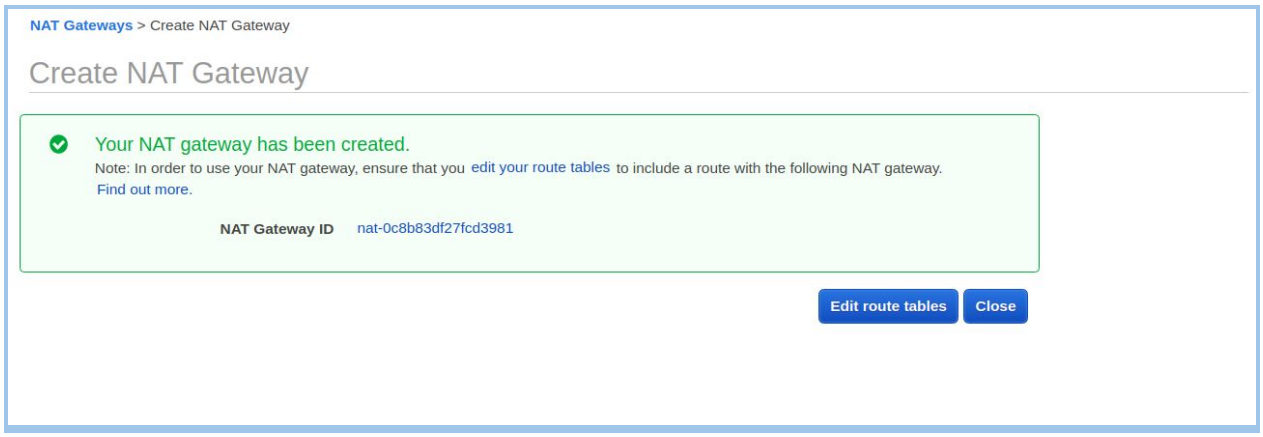
sudo apt-get install nginx -y

When the instance is launched, ssh and check if nginx is running.



Create a NAT gateway in the public subnet and attach it to the private subnet route table.

(NAT gateway uses an elastic IP)



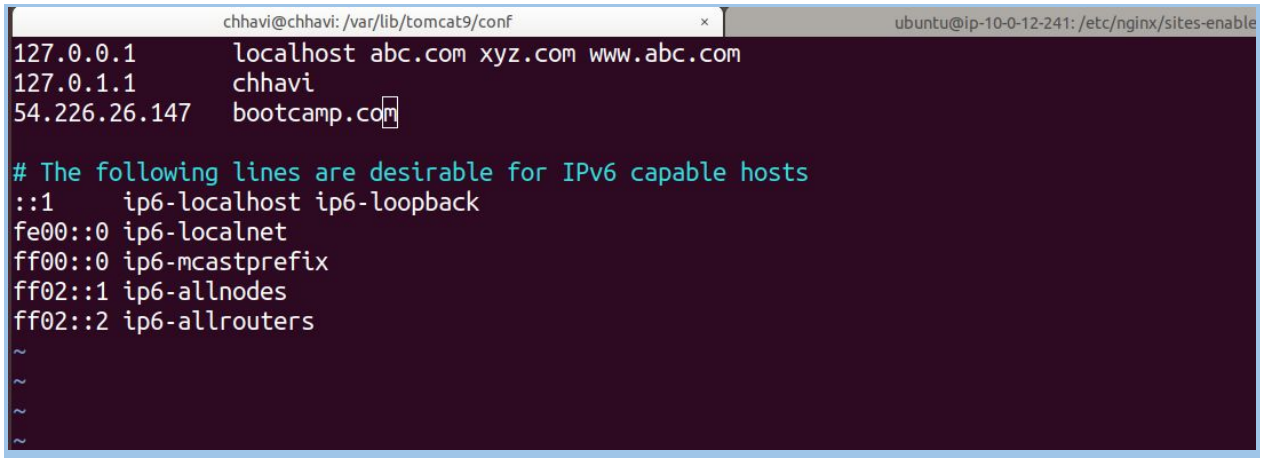
PROXY PASS

In the public instance create a virtual host. In the server block write the following.

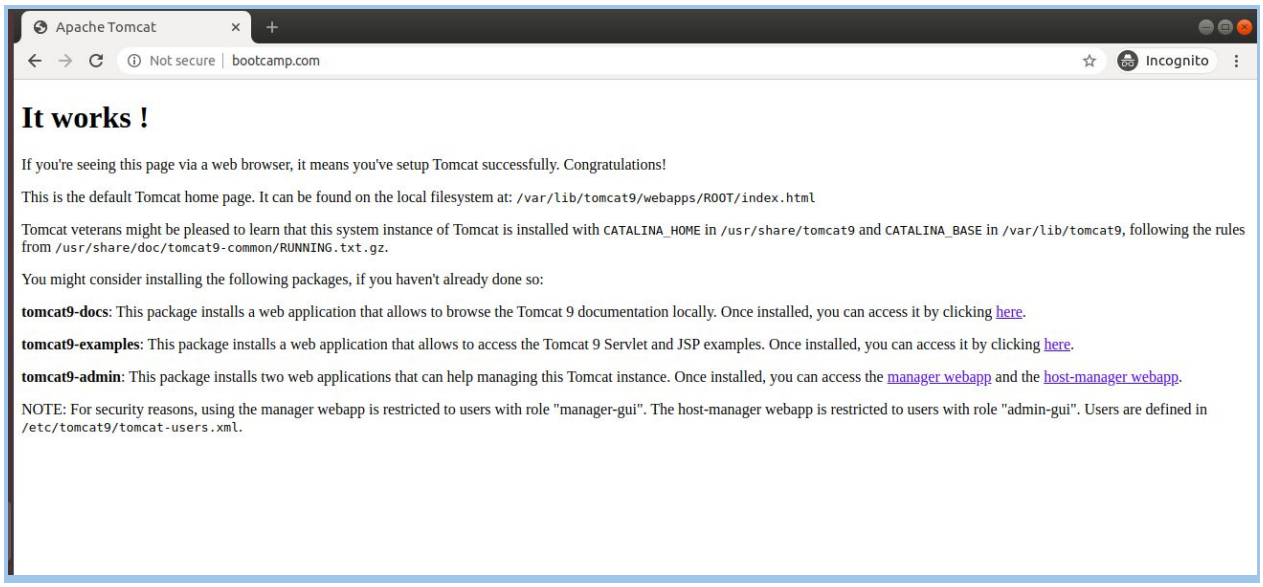
Proxy pass to the tomcat running in the private subnet.

```
server{

        listen 80;
        server_name bootcamp.com;
        location /{
                proxy_pass http://10.0.10.214:8080
}
}
~
~
```

Create an entry for the public IP and associated server name in the /etc/hosts file.

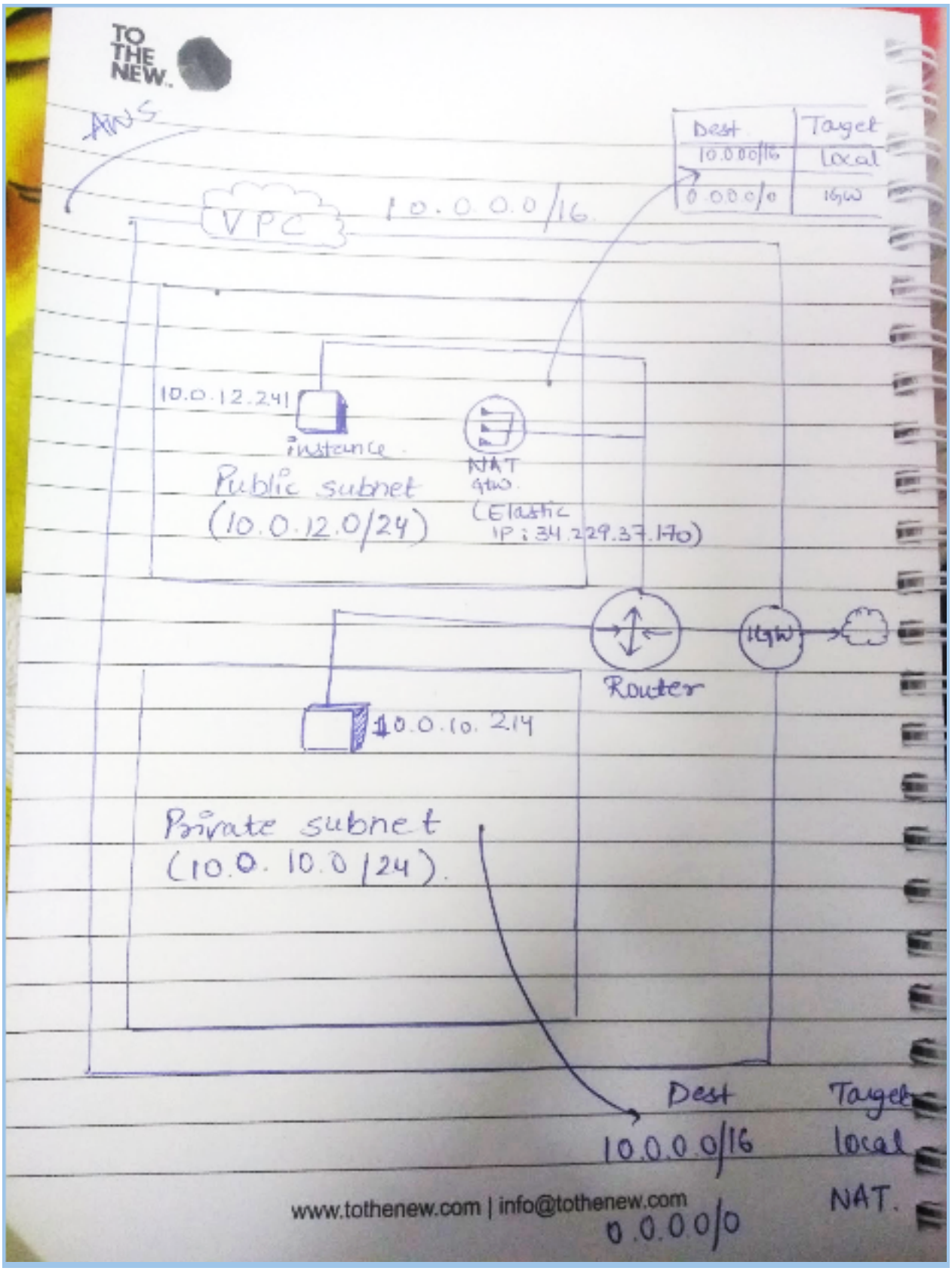


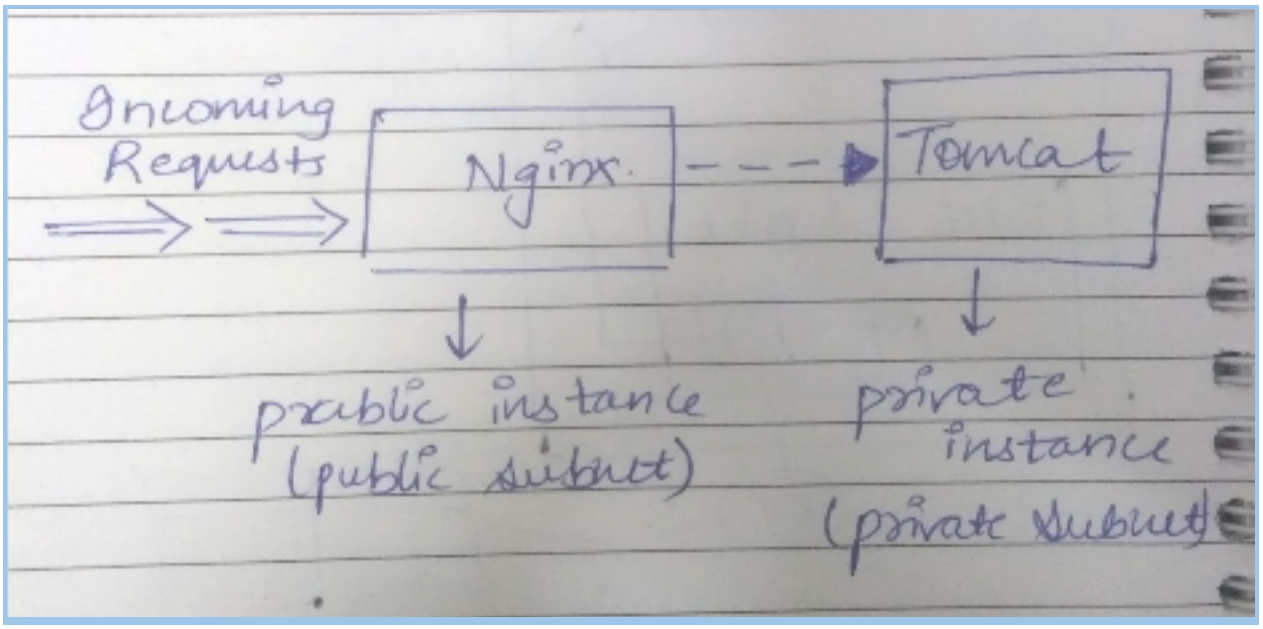
Hit bootcamp.com in the browser. You'll be proxy passed to Tomcat.



After Implementing this on AWS, create an architecture diagram for this use case.

Ans.


TO THE NEW.
AWS
VPC
10.0.0.0/16
Dest
Target
10.0.0.0/16
local
0.0.0.0/0
IGW
10.0.12.241
instance
Public subnet
(10.0.12.0/24)
NAT gtw.
(Elastic IP: 34.229.37.170)
Router
IGW
10.0.10.214
Private subnet
(10.0.10.0/24).
Dest
Target
10.0.0.0/16
local
0.0.0.0/0
NAT.
www.tothenew.com | info@tothenew.com


Incoming Requests
Nginx
Tomcat
public instance (public subnet)
private instance (private subnet)