

VPC Assignment

Q1. When to use Elastic IP over Public IP

Q2. Valid IP Ranges for LAN, Implication of using Public IP ranges for Private Network.

Q3. List down the things to keep in mind while VPC peering.

Q4. 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.

Q5. Differentiate between NACL and Security Groups.

Q6. Implement a 2-tier vpc with following requirements:

1. Create a private subnet, attach NAT, and host an application server(Tomcat)
2. Create a public subnet, and host a web server(Nginx), also proxypass to Tomcat from Nginx

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

Note: For hosting Nginx in public subnet, use Elastic IP.

Q1. When to use Elastic IP over Public IP

Elastic IP is used when you are working on long time project and configuration of IP sometimes consumes more time and you don't want your IP to change.

Q2. Valid IP Ranges for LAN, Implication of using Public IP ranges for Private Network.

Valid IP ranges of LAN:

- **192.168.0.0 - 192.168.255.255** (65,536 IP addresses)
- **172.16.0.0 - 172.31.255.255** (1,048,576 IP addresses)
- **10.0.0.0 - 10.255.255.255** (16,777,216 IP addresses)

It is public global addresses that are used in the Internet. A public IP address is an IP address that is used to access the Internet. Public (global) IP addresses are routed on the Internet, unlike private addresses.

The presence of a public IP address on your private network will allow you to organize your own server (VPN, FTP, WEB, etc.), remote access to your computer, video surveillance cameras, and access them from anywhere in the global network.

Q3. List down the things to keep in mind while VPC peering.

- Choosing the proper VPC configuration for your organization's needs
- Choosing a CIDR block for your VPC implementation
- Isolating your VPC environments
- Creating your disaster recovery plan
- Traffic control and security
- Keep your data close
- Determining the NAT instance type
- IAM for your AWS VPC infrastructure

Q4. 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.

No. of subnets created = $2^{\text{pow}4}=16$

No. of IPs in a subnet = $2^{\text{pow}12}=4096-2(\text{reserved})=4094$

Q5. Differentiate between NACL and Security Groups.

Security Group	NACL (Network Access Control List)
It supports only allow rules, and by default, all the rules are denied. You cannot deny the rule for establishing a connection.	It supports both allow and deny rules, and by default, all the rules are denied. You need to add the rule which you can either allow or deny it.

It is a stateful means that any changes made in the inbound rule will be automatically reflected in the outbound rule. For example, If you are allowing an incoming port 80, then you also have to add the outbound rule explicitly.	It is a stateless means that any changes made in the inbound rule will not reflect the outbound rule, i.e., you need to add the outbound rule separately. For example, if you add an inbound rule port number 80, then you also have to explicitly add the outbound rule.
It is associated with an EC2 instance.	It is associated with a subnet.
All the rules are evaluated before deciding whether to allow the traffic.	Rules are evaluated in order, starting from the lowest number.
Security Group is applied to an instance only when you specify a security group while launching an instance.	NACL has applied automatically to all the instances which are associated with an instance.
It is the first layer of defense.	It is the second layer of defense.

Q6. Implement a 2-tier vpc with following requirements:

- 1. Create a private subnet, attach NAT, and host an application server(Tomcat)**
- 2. Create a public subnet, and host a web server(Nginx), also proxypass to Tomcat from Nginx**

Answer

Step 1:- Create a VPC

[Create VPC](#) [Actions](#)

<< 1 to 5 of 5 >>

<input type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set
<input type="checkbox"/>	kaushlendra	vpc-007690f0f0127e1d5	available	10.0.0.0/16	-	dopt-519d6f34
<input type="checkbox"/>	sampurna_...	vpc-035e3e6135f2345ce	available	10.0.0.0/16	-	dopt-519d6f34
<input type="checkbox"/>	bijoy-vpc	vpc-07c3975194af4d40f	available	10.0.0.0/16	-	dopt-519d6f34
<input checked="" type="checkbox"/>	aman_vpc	vpc-0f6b3a34acb5c6306	available	10.3.0.0/16	-	dopt-519d6f34
<input type="checkbox"/>	default	vpc-d38d68b7	available	172.31.0.0/16	-	dopt-519d6f34

VPC: vpc-0f6b3a34acb5c6306

Description

CIDR Blocks

Flow Logs

Tags

VPC ID	vpc-0f6b3a34acb5c6306	Tenancy	default
State	available	Default VPC	No
IPv4 CIDR	10.3.0.0/16	Classic link	Disabled
IPv6 CIDR	-	IPv6 Pool	-
DNS resolution	Enabled	Network ACL	acl-017241e449ee98770
DNS hostnames	Disabled	DHCP options set	dopt-519d6f34
ClassicLink DNS Support	Disabled	Route table	rtb-0da543d681993f2ce

[Subnets](#) > [Create subnet](#)

Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag

aman_private

VPC*

vpc-0f6b3a34acb5c6306

Availability Zone

us-east-1c

VPC CIDRs

CIDR	Status	Status Reason
10.3.0.0/16	associated	

IPv4 CIDR block*

10.3.1.0/24

* Required

[Cancel](#) [Create](#)

Step 2:- Create two Subnet (aman_private and aman_public)

Subnet ID : subnet-0a476267086a01567 Add filter

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
aman_private	subnet-0a476267086a01567	available	vpc-0f6b3a34acb5c6306 ...	10.3.1.0/24	251	-

Subnet: subnet-0a476267086a01567

Description Flow Logs Route Table Network ACL Tags Sharing

Subnet ID	subnet-0a476267086a01567	State	available
VPC	vpc-0f6b3a34acb5c6306 aman_vpc	IPv4 CIDR	10.3.1.0/24
Available IPv4 Addresses	251	IPv6 CIDR	-
Availability Zone	us-east-1c (use1-az2)	Route Table	rtb-0da543d681993f2ce
Network ACL	acl-017241e449ee98770	Default subnet	No
Auto-assign public IPv4 address	No	Auto-assign IPv6 address	No
Outpost ID	-	Owner	187632318301

Name : aman_public Add filter

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
aman_public	subnet-00e31894ba830d419	available	vpc-0f6b3a34acb5c6306 ...	10.3.2.0/24	251	-

Subnet: subnet-00e31894ba830d419

Description Flow Logs Route Table Network ACL Tags Sharing

Subnet ID	subnet-00e31894ba830d419	State	available
VPC	vpc-0f6b3a34acb5c6306 aman_vpc	IPv4 CIDR	10.3.2.0/24
Available IPv4 Addresses	251	IPv6 CIDR	-
Availability Zone	us-east-1c (use1-az2)	Route Table	rtb-0da543d681993f2ce
Network ACL	acl-017241e449ee98770	Default subnet	No
Auto-assign public IPv4 address	No	Auto-assign IPv6 address	No
Outpost ID	-	Owner	187632318301

Step 3:- Create route tables

[Route Tables](#) > Create route table

Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Name tag ⓘ

VPC* ⓘ

* Required

[Cancel](#) [Create](#)

Create route table

Actions

1 to 10 of 10

	Name	Route Table ID	Explicit subnet association	Edge associations	Main	VPC ID
<input checked="" type="checkbox"/>	aman_route	rtb-08f05f77c5e198c62	2 subnets	-	No	vpc-0f6b3a34acb5c6306
<input type="checkbox"/>	Vaibhav_pub	rtb-0cc0210981a5a36a0	subnet-05c84217cfb76812d	-	No	vpc-007690f0f0127e1d5

Route Table: rtb-08f05f77c5e198c62

Summary

Routes

Subnet Associations

Edge Associations

Route Propagation

Tags

Edit subnet associations

1 to 2 of 2

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-0a476267086a01567 aman_private	10.3.1.0/24	-
subnet-00e31894ba830d419 aman_public	10.3.2.0/24	-

Step 4:- Create Internet Gateway (aman_ig)

Create internet gateway

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Name tag ⓘ

* Required

Cancel Create

[Internet gateways](#) > Create internet gateway

Create internet gateway

✓ The following internet gateway was created:

Internet gateway ID [igw-027bbc7ace42573f4](#)

Close

Create internet gateway

Actions

	Name	ID	VPC	Owner
<input checked="" type="checkbox"/>	aman_ig	igw-027bbc7ace4...	-	187632318301

Delete internet gateway

Attach to VPC

Detach from VPC

Add/Edit Tags

Step 5:- Attach Internet Gateway to VPC

[Internet gateways](#) > Attach to VPC

Attach to VPC

Attach an internet gateway to a VPC to enable communication with the internet. Specify the VPC you would like to attach below.

VPC* 

► **AWS Command Line Interface command**

Step 6:- Configure route table for public subnet and connect to Internet Gateway.

aman_public

rtb-0df06ad913a8f11d3

subnet-00e31894ba830d419

-

No

vpc-0f6b3a34acb5c6306

Summary

Routes

Subnet Associations

Edge Associations

Route Propagation

Tags

Edit routes

View

All routes

Destination	Target	Status	Propagated
10.3.0.0/16	local	active	No
0.0.0.0/0	igw-027bbc7ace42573f4	active	No

Step 7:- Create a Public Instance (aman_public_EC2)

[1. Choose AMI](#) [2. Choose Instance Type](#) [3. Configure Instance](#) [4. Add Storage](#) [5. Add Tags](#) [6. Configure Security Group](#) [7. Review](#)

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	<input type="text" value="1"/>	Launch into Auto Scaling Group
Purchasing option	<input type="checkbox"/> Request Spot instances	
Network	<input type="text" value="vpc-0f6b3a34acb5c6306 aman_vpc"/> Create new VPC	
Subnet	<input type="text" value="subnet-00e31894ba830d419 aman_public us-east-1"/> Create new subnet 251 IP Addresses available	
Auto-assign Public IP	<input type="text" value="Enable"/>	
Placement group	<input type="checkbox"/> Add instance to placement group	
Capacity Reservation	<input type="text" value="Open"/> Create new Capacity Reservation	
IAM role	<input type="text" value="None"/> Create new IAM role	

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

Add filter
1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
aman_public_EC2	i-058c7d753d36d5af3	t2.micro	us-east-1c	running	Initializing	None

Instance: i-058c7d753d36d5af3 (aman_public_EC2)
 Public IP: 54.89.214.226

Description

Status Checks

Monitoring

Tags

Instance ID	i-058c7d753d36d5af3	Public DNS (IPv4)	-
Instance state	running	IPv4 Public IP	54.89.214.226
Instance type	t2.micro	IPv6 IPs	-
Finding	You may not have permission to access AWS Compute Optimizer.		
Private DNS	ip-10-3-2-227.ec2.internal	Elastic IPs	-
Private IPs	10.3.2.227	Availability zone	us-east-1c
Secondary private IPs	-	Security groups	aman_public_EC2. view inbound rules. view outbound rules
VPC ID	vpc-0f6b3a34acb5c6306 (aman_vpc)	Scheduled events	No scheduled events
		AMI ID	ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20200112 (ami-07ebfd5b3428b6f4d)

Step 8:- Create a Public Instance (aman_private_EC2)

Step 3: Configure Instance Details

Number of instances i

[Launch into Auto Scaling Group](#) i

Purchasing option i
☐ Request Spot instances

Network i

[Create new VPC](#)

Subnet i

[Create new subnet](#)

Auto-assign Public IP i

Placement group i
☐ Add instance to placement group

Capacity Reservation i

[Create new Capacity Reservation](#)

IAM role i

[Create new IAM role](#)

Shutdown behavior i

[Cancel](#)
[Previous](#)
[Review and Launch](#)
[Next: Add Storage](#)

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	AI
aman_private_EC2	i-0da69eff1e25a4edf	t2.micro	us-east-1c	running	Initializing	Nc

Description	Status Checks	Monitoring	Tags
Instance ID	i-0da69eff1e25a4edf	Public DNS (IPv4)	-
Instance state	running	IPv4 Public IP	-
Instance type	t2.micro	IPv6 IPs	-
Finding	You may not have permission to access AWS Compute Optimizer.	Elastic IPs	-
Private DNS	ip-10-3-1-74.ec2.internal	Availability zone	us-east-1c
Private IPs	10.3.1.74	Security groups	aman_private_EC2. view inbound rules. view outbound rules
Secondary private IPs		Scheduled events	No scheduled events
VPC ID	vpc-0f6b3a34acb5c6306 (aman_vpc)	AMI ID	ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20200112 (ami-07ebfd5b3428b614d)
Subnet ID	subnet-0a476267086a01567 (aman_private)	Platform	-
Network interfaces	eth0	IAM role	-

Step 9:- Connect to aman_public_EC2 instance through ssh and configure nginx in it.

```

ubuntu@ip-10-3-2-227: ~
File Edit View Search Terminal Help
aman@Aman-Khandelwal:~/Downloads$ ssh -i aman_khandelwal_key.pem ubuntu@54.89.214.226
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 Sat Feb 22 06:24:02 UTC 2020

System load:  0.0               Processes:    88
Usage of /:   13.6% of 7.69GB   Users logged in:  0
Memory usage: 15%              IP address for eth0: 10.3.2.227
Swap usage:   0%

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

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by

```

```
ubuntu@ip-10-3-2-227:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [15 kB]
```

```
ubuntu@ip-10-3-2-227:~$ sudo apt-get install nginx
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjpeg-turbo8
  libjpeg8 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libtiff5 libwebp6
  libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
```

Step 10:- Copy ssh key from local machine to aman_public_EC2 through SCP.

```
aman@Aman-Khandelwal:~/Downloads$ scp -i aman_khandelwal_key.pem aman_khandelwal_key.pem
ubuntu@54.89.214.226:/home/ubuntu
aman_khandelwal_key.pem 100% 1692 5.4KB/s 00:00
aman@Aman-Khandelwal:~/Downloads$
```

```
ubuntu@ip-10-3-2-227:~$ ls
aman_khandelwal_key.pem
ubuntu@ip-10-3-2-227:~$
```

Step 11:- Create NAT Gateway in Public Subnet.

[NAT Gateways](#) > Create NAT Gateway

Create NAT Gateway

Create a NAT gateway and assign it an Elastic IP address. [Learn more.](#)

Subnet* subnet-00e31894ba830d419  

Elastic IP Allocation ID* eipalloc-029188abb5e9adde6   [Allocate Elastic IP address](#)

* Required

[Cancel](#) [Create a NAT Gateway](#)

Create NAT Gateway



Your NAT gateway has been created.

Note: In order to use your NAT gateway, ensure that you [edit your route tables](#) to include a route with the following NAT gateway.
[Find out more.](#)

NAT Gateway ID [nat-0e9840f87bd58e417](#)

[Edit route tables](#)

[Close](#)

Step 12:- Configure Nat Gateway Route in route table for private subnet.

Filter by tags and attributes or search by keyword

Name	Route Table ID	Explicit subnet association	Edge associations	Main	VPC ID
aman_private	rtb-08f05f77c5e198c62	subnet-0a476267086a01567	-	No	vpc-0f6b3a34acb5c6306

Route Table: rtb-08f05f77c5e198c62

Summary Routes Subnet Associations Edge Associations Route Propagation Tags

Edit routes

View All routes

Destination	Target	Status	Propagated
10.3.0.0/16	local	active	No
0.0.0.0/0	nat-0e9840f87bd58e417	active	No

Step 13:- Connect to Public Instance (aman_public_EC2) through SSH and from their connect to private instance (aman_private_EC2) in the private subnet.


```
ubuntu@ip-10-3-2-227:~$ ssh -i aman_khandelwal_key.pem ubuntu@10.3.1.74
The authenticity of host '10.3.1.74 (10.3.1.74)' can't be established.
ECDSA key fingerprint is SHA256:VMZSJxeCIgWIy3uSep/QBI4GCvtXNQfJXVTtQ2FDDcQ.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.3.1.74' (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 Sat Feb 22 06:49:13 UTC 2020

```
System load:  0.0               Processes:            86
Usage of /:   13.8% of 7.69GB   Users logged in:     0
Memory usage: 17%              IP address for eth0: 10.3.1.74
Swap usage:   0%
```

```
* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
  https://ubuntu.com/livepatch
```

```
0 packages can be updated.
```

```
ubuntu@ip-10-3-1-74:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [746 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/universe Translation-en
```

Step 14:- Install and configure tomcat9 on private subnet.

```
ubuntu@ip-10-3-1-74:~$ 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 libecjclipse-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 pcsd
  libnss-mdns fonts-dejavu-extra fonts-ipafont-gothic fonts-ipafont-mincho
  fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic tomcat9-admin tomcat9-docs
```

```
ubuntu@ip-10-3-1-74:~$ curl 127.0.0.1:8080
<?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>CATALINA_HOME</code> in <code>/usr/share/tomcat9</code> and <code>CATALINA_BASE</code> in <code>/var/lib/tomcat9</code>, following the rules from <code>/usr/share/doc/tomcat9-common/RUNNING.txt.gz</code>.</p>
```

Step 15:- Configure Security Group of public instance to allow request on port no. 80, and for private instance to allow request on port 8080.

Group ID : sg-045b1ea51c7a277d1 Add filter 1 to 1 of 1

Name	Group ID	Group Name	VPC ID	Owner	Description
	sg-045b1ea51c7a277d1	aman_private_EC2	vpc-0f6b3a34acb5c6306	187632318301	launch-wizai

Security Group: sg-045b1ea51c7a277d1

Description Inbound Outbound Tags

Edit

Type	Protocol	Port Range	Source	Description
Custom TCP Rule	TCP	8080	10.3.2.0/24	
SSH	TCP	22	0.0.0.0/0	

Group Name : aman_public_EC2 Add filter

Name	Group ID	Group Name	VPC ID	Owner	Description
sg-058dce078997e7b14	aman_public_EC2	vpc-0f6b3a34acb5c6306	187632318301	launch-wizard-	

Security Group: sg-058dce078997e7b14

Description Inbound Outbound Tags

Edit

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	:::0	
SSH	TCP	22	0.0.0.0/0	

Step 16:- Configure nginx web server for proxy_pass request to the tomcat server at port 8080.

```
File Edit View Search Terminal Help
GNU nano 2.9.3 default Modified

root /var/www/html;

# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name _;

location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    #try_files $uri $uri/ =404;
    proxy_pass http://10.3.1.74:8080;
}

# pass PHP scripts to FastCGI server
#
#location ~ \.php$ {
#    include snippets/fastcgi-php.conf;
#}
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line


```
ubuntu@ip-10-3-2-227:~$ cd /etc/nginx/sites-available/
ubuntu@ip-10-3-2-227:/etc/nginx/sites-available$ sudo nano default
ubuntu@ip-10-3-2-227:/etc/nginx/sites-available$ sudo service nginx restart
ubuntu@ip-10-3-2-227:/etc/nginx/sites-available$
```

Step 17:- Now Open the public IP on browser.



It works !

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

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

Tomcat veterans might be pleased to learn that this system instance of Tomcat is installed with `CATALINA_HOME` in `/usr/share/tomcat9` and `CATALINA_BASE` in `/var/lib/tomcat9`, following the rules from `/usr/share/doc/tomcat9-common/RUNNING.txt.gz`.

You might consider installing the following packages, if you haven't already done so:

tomcat9-docs: This package installs a web application that allows to browse the Tomcat 9 documentation locally. Once installed, you can access it by clicking [here](#).

tomcat9-examples: This package installs a web application that allows to access the Tomcat 9 Servlet and JSP examples. Once installed, you can access it by clicking [here](#).

tomcat9-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the [manager webapp](#) and the [host-manager webapp](#).

NOTE: For security reasons, using the manager webapp is restricted to users with role "manager-gui". The host-manager webapp is restricted to users with role "admin-gui". Users are defined in `/etc/tomcat9/tomcat-users.xml`.

AWS Architecture



