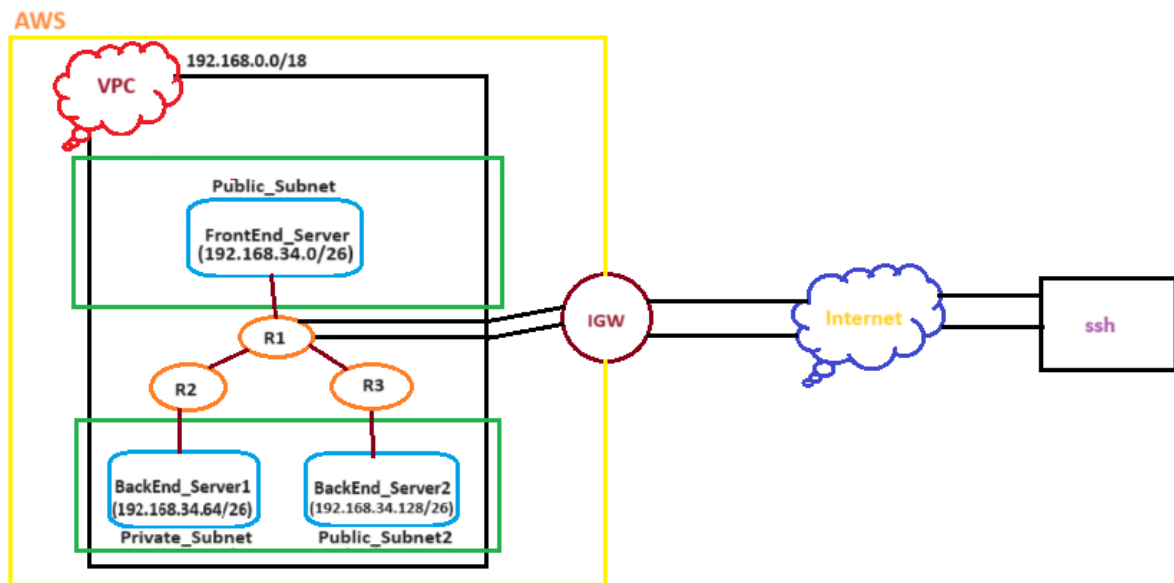


# Demonstrating VPC with 3 subnets to ping 3 instances with 3 routers with each other in same region

Architecture:



Steps:

- 1) Create a VPC and name it (My\_VPC) and assign a CIDR (192.168.0.0/18)

**VPC settings**

Resources to create [Info](#)  
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - *optional*  
Creates a tag with a key of 'Name' and a value that you specify.

My\_VPC

IPv4 CIDR block [Info](#)  
☒ IPv4 CIDR manual input ☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR  
192.168.0.0/18  
CIDR block size must be between /16 and /28.

IPv6 CIDR block [Info](#)  
☒ No IPv6 CIDR block ☐ IPAM-allocated IPv6 CIDR block ☐ Amazon-provided IPv6 CIDR block ☐ IPv6 CIDR owned by me

Tenancy [Info](#)  
Default

## Kavit Deepak Mehta

- 2) Now make 3 subnets, one public subnet (Public\_Subnet) and second & third private subnets (Private\_Subnet, Private\_Subnet2)

### Subnet 1 of 1

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

#### IPv4 VPC CIDR block [Info](#)

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

#### IPv4 subnet CIDR block

64 IPs

< > ^ v

### Subnet 1 of 1

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

#### IPv4 VPC CIDR block [Info](#)

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

#### IPv4 subnet CIDR block

64 IPs

< > ^ v

## Kavit Deepak Mehta

### Subnet 1 of 1

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

#### IPv4 VPC CIDR block [Info](#)

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

#### IPv4 subnet CIDR block

64 IPs

< > ^ v

- 3) Create an internet gateway and name it (My\_IGW) and also attach it your VPC (My\_VPC)

### Create internet gateway [Info](#)

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.

#### Internet gateway settings

##### Name tag

Creates a tag with a key of 'Name' and a value that you specify.

#### Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

##### Key

X

##### Value - optional

X

Remove

Add new tag

You can add 49 more tags.

Cancel

Create internet gateway

## Kavit Deepak Mehta

The following internet gateway was created: igw-0677af653224f890f - My\_IGW. You can now attach to a VPC to enable the VPC to communicate with the internet.

Attach to a VPC

[VPC](#) > [Internet gateways](#) > [Attach to VPC \(igw-0677af653224f890f\)](#)

### Attach to VPC (igw-0677af653224f890f) [Info](#)

**VPC**

Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs

Attach the internet gateway to this VPC.

► **AWS Command Line Interface command**

Cancel **Attach internet gateway**

- 4) Now create 3 route tables, 1 Public route tables and 2 Private route tables

### Create route table [Info](#)

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

**Route table settings**

Name - *optional*

Create a tag with a key of 'Name' and a value that you specify.

VPC

The VPC to use for this route table.

**Tags**

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - <i>optional</i>	
<input type="text" value="Name"/>	<input type="text" value="Public_Router"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

Cancel **Create route table**

## Create route table [Info](#)

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

### Route table settings

Name - *optional*

Create a tag with a key of 'Name' and a value that you specify.

Private\_Router1

VPC

The VPC to use for this route table.

vpc-0c79d2974fe192906 (My\_VPC) ▼

### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Q Name X

Value - *optional*

Q Private\_Router1 X

Remove

Add new tag

You can add 49 more tags.

Cancel

Create route table

## Create route table [Info](#)

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

### Route table settings

Name - *optional*

Create a tag with a key of 'Name' and a value that you specify.

Private\_Router2

VPC

The VPC to use for this route table.

vpc-0c79d2974fe192906 (My\_VPC) ▼

### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Q Name X

Value - *optional*

Q Private\_Router2 X

Remove

Add new tag

You can add 49 more tags.

Cancel

Create route table

- 5) Now in Public route table edit explicit subnet associations and associate Public Subnet and also edit route and add Interent Gateway Route and save it and in both Private route tables edit subnet assocaitions and associate Private Subnet to both of them and save it

rtb-09e3045747b3c299d / Public\_Router

Actions

Details

Info

Route table ID  
rtb-09e3045747b3c299d

VPC  
vpc-0c79d2974fe192906 | My\_VPC

Main  
No

Owner ID  
851725375246

Explicit subnet associations  
-

Edge associations  
-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Explicit subnet associations (0)

Edit subnet associations

Find subnet association

Name

Subnet ID

IPv4 CIDR

IPv6 CIDR

No subnet associations

You do not have any subnet associations.

VPC

Route tables

rtb-09e3045747b3c299d

Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)

Filter subnet associations

Name

Subnet ID

IPv4 CIDR

IPv6 CIDR

Route table ID

☒

Public\_Subnet

subnet-05237c4c5b05cbd86

192.168.34.0/26

-

Main (rtb-02049bb2ad4f899fb)

☐

Private\_Subnet

subnet-0382f185d78ea0fe0

192.168.34.64/26

-

Main (rtb-02049bb2ad4f899fb)

Selected subnets

subnet-05237c4c5b05cbd86 / Public\_Subnet

Cancel

Save associations

VPC

Route tables

rtb-09e3045747b3c299d

rtb-09e3045747b3c299d / Public\_Router

Actions

Details

Info

Route table ID  
rtb-09e3045747b3c299d

VPC  
vpc-0c79d2974fe192906 | My\_VPC

Main  
No

Owner ID  
851725375246

Explicit subnet associations  
subnet-05237c4c5b05cbd86 /  
Public\_Subnet

Edge associations  
-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (1)

Both

Edit routes

Filter routes

Destination

Target

Status

Propagated

192.168.0.0/18

local

Active

No

# Kavit Deepak Mehta

VPC > Route tables > rtb-09e3045747b3c299d > Edit routes

### Edit routes

Destination	Target	Status	Propagated
192.168.0.0/18	local	Active	No
0.0.0.0/0	Internet Gateway	-	No

[Add route](#) [Cancel](#) [Preview](#) [Save changes](#)

VPC > Route tables > rtb-036d07efd384f0804 > Edit subnet associations

### Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
Public_Subnet	subnet-05237c4c5b05cbd86	192.168.34.0/26	-	rtb-09e3045747b3c299d / Public_Router
Private_Subnet	subnet-0382f185d78ea0fe0	192.168.34.64/26	-	Main (rtb-02049bb2ad4f899fb)

Selected subnets

subnet-0382f185d78ea0fe0 / Private\_Subnet

[Cancel](#) [Save associations](#)

VPC > Route tables > rtb-09c2e2fcfb25713f1 > Edit subnet associations

### Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/3)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
Public_Subnet	subnet-05237c4c5b05cbd86	192.168.34.0/26	-	rtb-09e3045747b3c299d / Public_Router
Private_Subnet	subnet-0382f185d78ea0fe0	192.168.34.64/26	-	rtb-09c2e2fcfb25713f1 / Private_Rout...
Private_Subnet2	subnet-04c2d5546b8574314	192.168.34.128/26	-	Main (rtb-02049bb2ad4f899fb)

Selected subnets

subnet-04c2d5546b8574314 / Private\_Subnet2



[Cancel](#) [Save associations](#)

## 6) Now create 2 instances, 1 public instance and 2 private instances



Instances (3) <a href="#">Info</a>								<a href="#">Refresh</a> <a href="#">Connect</a> <a href="#">Instance state</a> <a href="#">Actions</a> <a href="#">Launch instances</a>	
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>								<a href="#">All states</a>	
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone		
<input type="checkbox"/>	FrontEnd_Server	i-088baa7ef7b78ba82	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>	ap-south-1a		
<input type="checkbox"/>	BackEnd_Server2	i-07ca23495aa39b311	Running	t2.micro	Initializing	<a href="#">View alarms</a>	ap-south-1b		
<input type="checkbox"/>	BackEnd_Server1	i-0c8e217472fda9209	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>	ap-south-1b		

7) In Inbound Rules of every instance, Add an ICMP of other 2 instances  
Subnet IP Address and vice versa



## a) FrontEnd\_Server

Inbound rules (3)								Manage tags	Edit inbound rules
<input type="text" value="Search"/>							< 1 > 		
Security group rule...	IP version	Type	Protocol	Port range	Source				
sgr-002cd65da184b70...	IPv4	SSH	TCP	22	0.0.0.0/0				
sgr-00c440dacf004eb6d	IPv4	All ICMP - IPv4	ICMP	All	192.168.34.64/26				
sgr-080044dab926c69f2	IPv4	All ICMP - IPv4	ICMP	All	192.168.34.128/26				

## b) BackEnd\_Server1

Inbound rules (3)								Manage tags	Edit inbound rules
<input type="text" value="Search"/>							< 1 > 		
Security group rule...	IP version	Type	Protocol	Port range	Source				
sgr-03bbffc98dd1f059b	IPv4	All ICMP - IPv4	ICMP	All	192.168.34.0/26				
sgr-02b81e1df8de0ad8f	IPv4	SSH	TCP	22	0.0.0.0/0				
sgr-03e51366598d9f9...	IPv4	All ICMP - IPv4	ICMP	All	192.168.34.128/26				

## c) BackEnd\_Server2

Inbound rules (3)								Manage tags	Edit inbound rules
<input type="text" value="Search"/>							< 1 > 		
Security group rule...	IP version	Type	Protocol	Port range	Source				
sgr-04939b9398d6b5...	IPv4	SSH	TCP	22	0.0.0.0/0				
sgr-0d0553c9d4c617ffb	IPv4	All ICMP - IPv4	ICMP	All	192.168.34.0/26				
sgr-0d84f0fbf637ff54e	IPv4	All ICMP - IPv4	ICMP	All	192.168.34.64/26				



- 8) Now connect the 1<sup>st</sup> instance (FrontEnd\_Server) and ping other 2 private instances (BackEnd\_Server1, BackEnd\_Server2) and vice versa

```
ec2-user@ip-192-168-34-47:~ × + ▾
```

```
Microsoft Windows [Version 10.0.22631.3737]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ITSKDM>ssh -i "C:\Users\ITSKDM\Downloads\LinuxKey.pem" ec2-user@65.0.184.254

#_
~\_ #####_ Amazon Linux 2023
~~ \#####\
~~ \###|
~~ \|/_ --- https://aws.amazon.com/linux/amazon-linux-2023
~~ V~' '~>
    ~~~
      ~.. _./ _./
        _/m/'
```

```
[ec2-user@ip-192-168-34-47 ~]$ ping 192.168.34.96
PING 192.168.34.96 (192.168.34.96) 56(84) bytes of data.
64 bytes from 192.168.34.96: icmp_seq=1 ttl=127 time=1.16 ms
64 bytes from 192.168.34.96: icmp_seq=2 ttl=127 time=0.717 ms
64 bytes from 192.168.34.96: icmp_seq=3 ttl=127 time=0.738 ms
^Z
[1]+ Stopped                  ping 192.168.34.96
[ec2-user@ip-192-168-34-47 ~]$ ping 192.168.34.77
PING 192.168.34.77 (192.168.34.77) 56(84) bytes of data.
64 bytes from 192.168.34.77: icmp_seq=1 ttl=127 time=1.14 ms
64 bytes from 192.168.34.77: icmp_seq=2 ttl=127 time=0.945 ms
64 bytes from 192.168.34.77: icmp_seq=3 ttl=127 time=0.920 ms
^Z
[2]+ Stopped                  ping 192.168.34.77
[ec2-user@ip-192-168-34-47 ~]$
```

[illegible]

```
#_
~\_##### Amazon Linux 2023
~~\#####\
~~\###|
~~\#/ --- https://aws.amazon.com/linux/amazon-linux-2023
~~V~' '->
    ~~~
      ~~-.-./ -/\
        _/_/' -/\
          _/m/'
```

[ec2-user@ip-192-168-34-184 ~]\$ ping 192.168.34.47  
PING 192.168.34.47 (192.168.34.47) 56(84) bytes of data.  
64 bytes from 192.168.34.47: icmp\_seq=1 ttl=127 time=1.09 ms  
64 bytes from 192.168.34.47: icmp\_seq=2 ttl=127 time=0.780 ms  
64 bytes from 192.168.34.47: icmp\_seq=3 ttl=127 time=0.717 ms  
^Z  
[1]+ Stopped ping 192.168.34.47  
[ec2-user@ip-192-168-34-184 ~]\$ ping 192.168.34.96  
PING 192.168.34.96 (192.168.34.96) 56(84) bytes of data.  
64 bytes from 192.168.34.96: icmp\_seq=1 ttl=127 time=0.336 ms  
64 bytes from 192.168.34.96: icmp\_seq=2 ttl=127 time=0.519 ms  
64 bytes from 192.168.34.96: icmp\_seq=3 ttl=127 time=0.510 ms  
^Z  
[2]+ Stopped ping 192.168.34.96  
[ec2-user@ip-192-168-34-184 ~]\$ |