

Project Title : Establishing VPC Peering Between Two VPCs in AWS

1. Introduction

This project outlines the process of establishing a VPC peering connection between two Virtual Private Clouds (VPCs) in AWS, enabling network traffic between instances in different VPCs.

2. Objectives

- Create two VPCs with distinct CIDR blocks.
- Establish a VPC peering connection between them.
- Configure route tables to allow communication.
- Verify connectivity between instances in the peered VPCs.

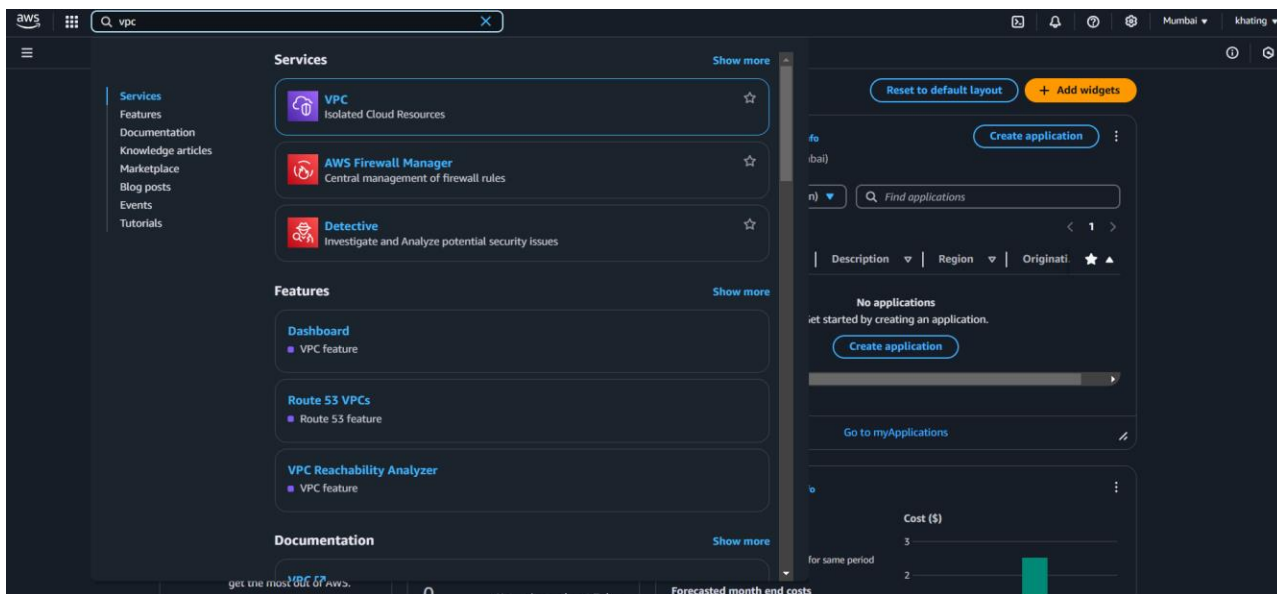
3. Prerequisites

- An AWS account with necessary permissions.
- Basic understanding of AWS networking concepts.
- Access to the AWS Management Console.

4. Steps to Implement

Step 1: Create VPCs

Navigate to the VPC Dashboard : In the AWS Management Console, go to the VPC service.



Create the First VPC : Click on "Create VPC".

Set the following details :

- Name tag: Production_VPC
- IPv4 CIDR block: 10.0.0.0/16
- Leave other settings as default and click "Create VPC".

Create the Second VPC Repeat the above steps with:

Name tag: Operations_VPC

IPv4 CIDR block: 192.168.0.0/16

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.

Production_VPC

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

10.0.0.0/16

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

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 - optional	
Q Name	Q Production_VPC	Remove tag

[Add tag](#)

You can add 49 more tags

[Cancel](#) [Preview code](#) [Create VPC](#)

Step 2: Create Subnets

1. Add a Subnet to Production_VPC

Select PRODUCTION_VPC and click on "Create Subnet".

Configure:

- Subnet name: prod_public_subnet
- Availability Zone: Choose one (e.g., ap-south-1a).
- IPv4 CIDR block: 10.0.0.0/24

Click "Create Subnet".

2. Add a Subnet to OPERATIONS_VPC

- Repeat the above steps for OPERATIONS_VPC with:
 - Subnet name: Ops_public_Subnet
 - Availability Zone: Choose one (e.g., ap-south-1a).
 - IPv4 CIDR block: 192.168.0.0/24

Create subnets in this VPC.

vpc-0b43cf52a3475c661 (Production_VPC) ▼

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
prod_public_subnet
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.
Asia Pacific (Mumbai) / ap-south-1a ▼

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.
10.0.0.0/16 ▼

IPv4 subnet CIDR block
10.0.0.0/24 256 IPs
< > ^ v

▼ **Tags - optional**

Key	Value - optional	
Q Name X	Q prod_public_subnet X	Remove

Step 3: Create and Attach Internet Gateway to your VPC

- Click on Create internet gateway.
- Provide a Name tag for easier identification, e.g., Prod_IGW.
- Click Create internet gateway to proceed.

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.
Prod_IGW

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	Value - optional	
Q Name X	Q Prod_IGW X	Remove

[Add new tag](#)
You can add 49 more tags.

Cancel [Create internet gateway](#)

- In the dialog box, select the VPC you want to attach the IGW to.
- Click on Actions, then choose Attach to VPC.

- Click Attach internet gateway to complete the attachment.

Attach to VPC (igw-083ecba55332cf2b8) 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

Step 4: Associate the Route Table with the Subnet & Edit the Route Table to Use the Internet Gateway

1. Navigate to Subnet Associations:

- In the Route Tables section, select the route table you've just updated.
- Go to the Subnet associations tab.
- Click Edit subnet associations.

Edit subnet associations
Change which subnets are associated with this route table.

Available subnets (1/1)

<input checked="" type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	prod_public_subnet	subnet-00dda03140ee89ceb	10.0.0.0/24	-	Main (rtb-082a5f7fdec76dbf5 / prod_

Selected subnets

subnet-00dda03140ee89ceb / prod_public_subnet ×

Cancel Save associations

- Select the subnet(s) you want to associate with this route table.
- Click Save associations to confirm.
-

1. Edit Route Tables to use the internet gateway

- In the VPC Dashboard, select Route Tables from the left-hand menu.
- Identify and select the route table associated with the subnet you wish to provide internet access to.
- Click Edit routes, then Add route.
- In the Destination field, enter 0.0.0.0/0 to represent all IPv4 addresses.
- In the Target field, select the Internet Gateway ID (e.g., igw-xxxxxxx).
- Click Save routes to apply the changes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway igw-083ecba55332cf2b8	-	No

Buttons: Add route, Remove, Cancel, Preview, Save changes

Step 5 : Launch an EC2 Instance in the Public Subnet of Both the VPC.

1. Open the Amazon EC2 Console.
2. Click Launch Instance.
3. Provide an instance name (e.g., My_EC2_Instance).
4. Select an Amazon Machine Image (AMI) (e.g., Amazon Linux 2 AMI)
5. Choose an instance type (e.g., t2.micro).
6. Under Key pair (login), select an existing key pair or create a new one.
7. In Network settings, select your VPC (e.g., Production_VPC) and the public subnet (e.g., prod_public_subnet).

Name

prod_server

Add additional tags

▼ Application and OS Images (Amazon Machine Image)
Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents
Quick Start

Amazon Linux
aws

macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Linux
SUSE Linux

t

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI
ami-0fd05997b4dff7aac (64-bit (x86), uefi-preferred) / ami-013b2876e77b2db31 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible
▼

Description

Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.6.20241212.0 x86_64 HVM kernel-6.1

Architecture
64-bit (x86)
▼

Boot mode
uefi-preferred

AMI ID
ami-0fd05997b4dff7a

Username
ec2-user

Verified provider

- Ensure Auto-assign public IP is enabled.
- Select an existing security group or create a new one with appropriate inbound rules (e.g., allowing SSH access).
- Click Launch instance.

VPC - required
Info

vpc-0b43cf52a3475c661 (Production_VPC)
10.0.0.0/16

Create new VPC

Subnet
Info

subnet-00dda03140ee89ceb
prod_public_subnet

VPC: vpc-0b43cf52a3475c661 Owner: 851725487902
Availability Zone: ap-south-1a Zone type: Availability Zone
IP addresses available: 251 CIDR: 10.0.0.0/24

Create new subnet

Auto-assign public IP
Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups)
Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

Security group name - required

MYSG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&:!\$*

Description - required
Info

launch-wizard-1 created 2024-12-27T06:24:33.170Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)
Remove

Type
Info
ssh
▼

Protocol
Info
TCP

Port range
Info
22

Source type
Info
Anywhere
▼

Source
Info
Add CIDR, prefix list or security g

Description - optional
Info
e.g. SSH for admin desktop

Step 6: Create a VPC Peering Connection

- In the VPC dashboard, select "Peering Connections".
- Click "Create Peering Connection".
- Set:
 - Peering connection name tag: PRODUCTION_VPC-to-OPERATIONS_VPC
 - Requester VPC: PRODUCTION_VPC
 - Acceptor VPC: OPERATIONS_VPC
- Click "Create Peering Connection".

Create peering connection

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them privately. [Info](#)

Peering connection settings

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

Select a local VPC to peer with

VPC ID (Requester)

VPC CIDRs for vpc-0b43cf52a3475c661 (Production_VPC)

CIDR	Status	Status reason
10.0.0.0/16	✔ Associated	-

Select another VPC to peer with

Account
☒ My account
☐ Another account

Region
☒ This Region (ap-south-1)
☐ Another Region

VPC ID (Acceptor)

VPC CIDRs for vpc-05033a8738ddea4e7 (Operations_VPC)

CIDR	Status	Status reason
192.168.0.0/16	✔ Associated	-

- In "Peering Connections", select the newly created connection.
- Click "Actions" > "Accept Request".
- Confirm acceptance.

✔ A VPC peering connection pcx-0553097dbd581a314 / MypeeringX has been requested. ✕

Peering connections (1/1) [Info](#)

Name	Peering connection ID	Status	Requester VPC	Acceptor VPC	Requester
MypeeringX	pcx-0553097dbd581a314	⌚ Pending acceptance	vpc-0b43cf52a3475c661 / Prod...	vpc-05033a8738ddea4e7 / Op...	10.0.0.0/16

Actions

- View details
- Accept request
- Reject request
- Edit DNS settings
- Manage tags
- Delete peering connection

Step 7: Update Route Tables

1. Modify Route Table for PRODUCTION_VPC

- Navigate to "Route Tables".
- Select the route table associated with PRODUCTION_VPC.
- Click "Edit Routes" and add:
 - Destination: 192.168.0.0/16
 - Target: Select the peering connection PRODUCTION_VPC-to-OPERATIONS_VPC
- Save changes.

2. Modify Route Table for OPERATIONS_VPC

- Repeat the above steps for OPERATIONS_VPC with:
 - Destination: 10.0.0.0/16
 - Target: Select the peering connection PRODUCTION_VPC-to-OPERATIONS_VPC

Edit routes

Destination	Target	Status	Propagated	
10.0.0.0/16	local	Active	No	
<input type="text" value="192.168.0.0/16"/>	<input type="text" value="local"/>			
	Peering Connection	Active	No	Remove
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="pcx-0553097dbd581a314"/>			
	Internet Gateway	Active	No	Remove
	<input type="text" value="igw-083ecba55332cf2b8"/>			
<button>Add route</button>				
				<button>Cancel</button> <button>Preview</button> <button>Save changes</button>

Edit routes

Destination	Target	Status	Propagated	
192.168.0.0/16	local	Active	No	
<input type="text" value="10.0.0.0/16"/>	<input type="text" value="local"/>			
	Peering Connection	Active	No	Remove
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="pcx-0553097dbd581a314"/>			
	Internet Gateway	Active	No	Remove
	<input type="text" value="igw-0d2e6aa15d07d6294"/>			
<button>Add route</button>				
				<button>Cancel</button> <button>Preview</button> <button>Save changes</button>

Step 8 : Configure Security Groups

1. Update Security Group for Instances in PRODUCTION_VPC

- Ensure the security group allows inbound traffic from SSH via port 22 from Anywhere and also Allows traffic from ICMP(Internet control messaging protocol) from Anywhere.

2. Update Security Group for Instances in OPERATIONS_VPC


- Ensure the security group allows inbound traffic from SSH via port 22 from Anywhere and also Allows traffic from ICMP(Internet control messaging protocol) from Anywhere.

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>	
sgr-0ce6ece9f8bd161b8	SSH	TCP	22	Custom	<input type="text" value="Q"/>	<input type="button" value="Delete"/>
					<input type="text" value="0.0.0.0/0"/>	
sgr-0f1265e9fe19b84ff	All ICMP - IPv4	ICMP	All	Custom	<input type="text" value="Q"/>	<input type="button" value="Delete"/>
					<input type="text" value="0.0.0.0/0"/>	

 Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Step 9 : Launch EC2 Instances and Test Connectivity

1. Test Connectivity

- Use ping or SSH to test connectivity between the instances across VPCs.

SSH into the Instances

2. Open a terminal on your local machine.

3. Connect to the instances using SSH: `ssh -i mykey.pem ec2-user@<public ip address>`

-


```
ec2-user@ip-10-0-0-46:~$ ssh -i mykey.pem ec2-user@52.66.52.173
#_
~\_ #####_ Amazon Linux 2023
nn \_#####\
nn \_###|
nn \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
3
nn V~' '->
nn
nn _.'
nn _/_/_/
nn _/m/'

Last login: Fri Dec 27 06:38:04 2024 from 103.48.101.180
[ec2-user@ip-10-0-0-46 ~]$ ping 192.168.0.246
PING 192.168.0.246 (192.168.0.246) 56(84) bytes of data.
64 bytes from 192.168.0.246: icmp_seq=1 ttl=127 time=1.34 ms
64 bytes from 192.168.0.246: icmp_seq=2 ttl=127 time=0.576 ms
64 bytes from 192.168.0.246: icmp_seq=3 ttl=127 time=0.371 ms
64 bytes from 192.168.0.246: icmp_seq=4 ttl=127 time=0.764 ms
64 bytes from 192.168.0.246: icmp_seq=5 ttl=127 time=1.24 ms
64 bytes from 192.168.0.246: icmp_seq=6 ttl=127 time=0.546 ms
64 bytes from 192.168.0.246: icmp_seq=7 ttl=127 time=0.754 ms
64 bytes from 192.168.0.246: icmp_seq=8 ttl=127 time=0.390 ms
64 bytes from 192.168.0.246: icmp_seq=9 ttl=127 time=0.436 ms
64 bytes from 192.168.0.246: icmp_seq=10 ttl=127 time=0.351 ms
64 bytes from 192.168.0.246: icmp_seq=11 ttl=127 time=0.329 ms
64 bytes from 192.168.0.246: icmp_seq=12 ttl=127 time=0.739 ms
^C
--- 192.168.0.246 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11380ms
rtt min/avg/max/mdev = 0.329/0.653/1.341/0.323 ms
[ec2-user@ip-10-0-0-46 ~]$
```