

VPC PEERING CONNECTION

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses. Instances in either VPC can communicate with each other as if they are within the same network.

You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account. The VPCs can be in different Regions or the same Region.

Types of VPC Peering

<u>Inter-region VPC Peering</u>	<u>Cross-account VPC Peering</u>
<ul style="list-style-type: none">Connects VPCs in different cloud regions.Enables global communication between resources across regions.Helps in building distributed applications while maintaining network isolation.	<ul style="list-style-type: none">Connects VPCs owned by different AWS accounts.Facilitates collaboration or resource sharing between different accounts.Requires explicit authorization from both AWS accounts involved.Ensures network segregation and security between peered VPCs.

To Create a VPC Peering Connection follow the following steps.

Step 1) Create two VPCs in two different Regions. Ensure that the CDIR block of both VPC does not overlap.

First VPC in the Paris Region

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.

VPC-1-Paris

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input

☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

10.10.0.0/20

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

Second VPC in the London Region

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.

VPC-2-London

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

IPv4 CIDR
20.20.0.0/24
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

Step 2) After Creating VPC we need to create subnets for both VPC.

- In Over Paris Region VPC, we will create **1 public subnet** and **1 private subnet**

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

Public-Subnet-Paris
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.

Europe (Paris) / eu-west-3c

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.

10.10.0.0/20

IPv4 subnet CIDR block
10.10.0.0/21 2,048 IPs

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

Private-Subnet-Paris
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.

Europe (Paris) / eu-west-3c

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.

10.10.0.0/20

IPv4 subnet CIDR block
10.10.8.0/24 256 IPs

- After creating a subnet, go to the public subnet setting, enable Auto-assign Public IP (to make public this subnet), and save the changes.

This screenshot shows the 'Auto-assign IP settings' for a subnet named 'Public-subnet-Paris'. The 'Enable auto-assign public IPv4 address' checkbox is checked, which is the required step to make the subnet public.

- Now Create a Subnet in the London Region. Here we create the only Private subnet.

This screenshot shows the 'Create subnet' form. The 'Subnet name' is 'Private-Subnet-London'. The 'Availability Zone' is set to 'Europe (London) / eu-west-2a'. The 'IPv4 VPC CIDR block' is '20.20.0.0/24'. The 'IPv4 subnet CIDR block' is '20.20.0.0/25', which provides 128 IP addresses.

Step 3) Now We Need to Create an Internet Gateway for VPC in the Paris Region.

- Create Internet Gateway and save.

This screenshot shows the 'Create internet gateway' form. The 'Name tag' is 'Paris-IGW'. The description states: '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.'

- After Creating IGW attach it to our custom VPC in Paris Region.

This screenshot shows the 'Attach to VPC' form for the internet gateway 'igw-0281fc1164d0d8565'. The 'Available VPCs' section shows a search for 'vpc-09f18dacc5ae18a9'. The 'AWS Command Line Interface command' is also visible. At the bottom, there are 'Cancel' and 'Attach internet gateway' buttons.

Step 4) When we created the VPC, the route table was created by default.

- Edit the default Route Table and add the route setting up a target to the Internet gateway.

VPC > Internet gateways > Attach to VPC (igw-0281fc1164d0d8565)

Attach to VPC (igw-0281fc1164d0d8565) [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 5) Now we will create a peering Request from our VPC in the Paris region for VPC in the London region.

- In VPC Dashboard select Peering connection Option, and click on the Create Peering connection button.

EC2 S3 EFS VPC IAM Console Home

Endpoint services
NAT gateways
Peering connections
Security
Network ACLs

Peering connections [Info](#)

[Create peering connection](#)

Name	Peering connection ID	Status	Requester VPC	Accepter VPC
No peering connection found				

- Name Peering connection, and select our Custom VPC ID.

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-09f18dacc5ae18a9 (VPC-1-Paris)

CIDR	Status	Status reason
10.10.0.0/20	Associated	-

- Now Select the London region where we created our 2nd VPC, copy the London region's Custom VPC ID, then paste it here in the VPC ID acceptor box, and click ok Create Peering connection button.

Select another VPC to peer with

Account
☒ My account
☐ Another account

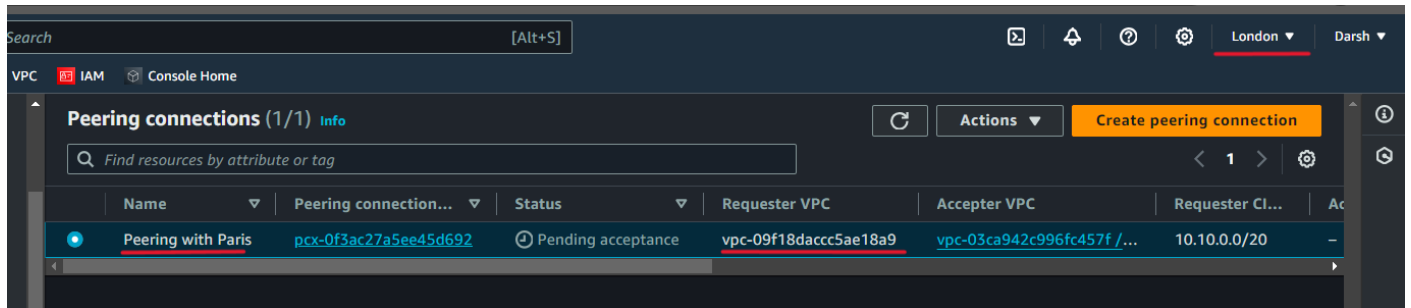
Region
☐ This Region (eu-west-3)
☒ Another Region

VPC ID (Acceptor)

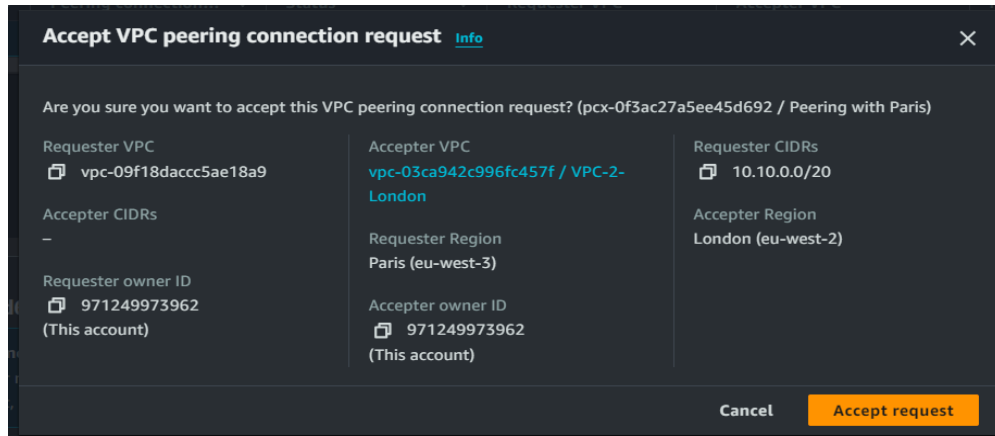
Our VPC peering connection request is created.

Step 6) Now we need to accept the peering connection request in the London region.

- Go to the peering connection in the London region's VPC dashboard, you will see the peering request.



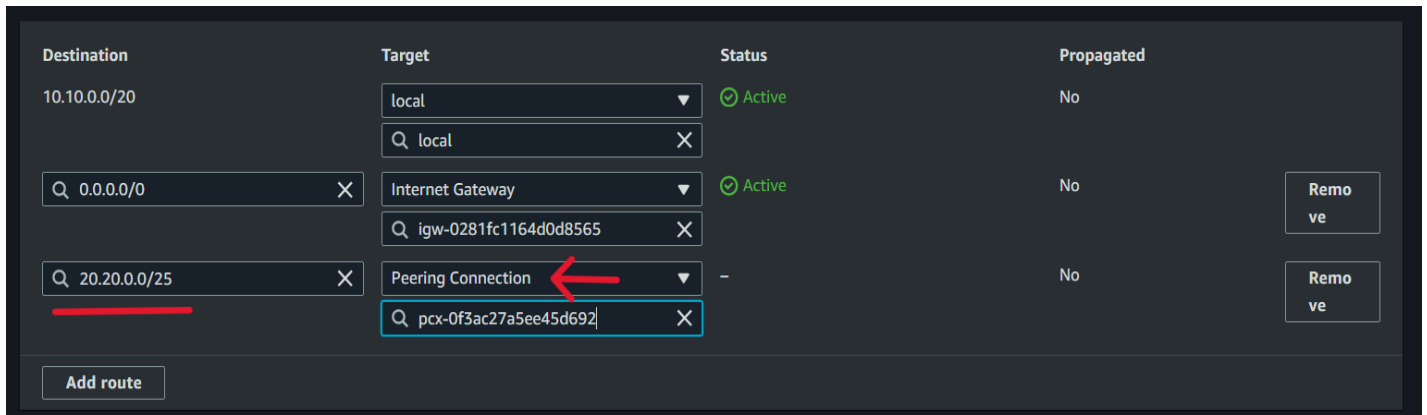
- Click on the Action button and accept it.



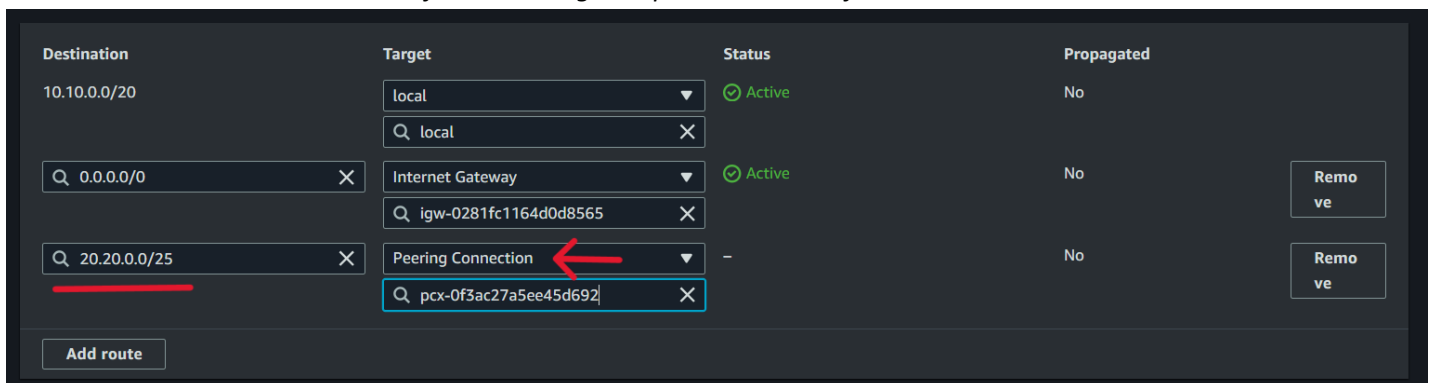
Here is our peering connection created.

Step 7) Now we need to add a route target for the VPC peering connection in the Route table of both VPCs.

- First, we will add a route in the Paris region VPC Route table.
- Copy the CIDR of the London region's private subnet of our custom VPC.
- Edit the Route Table of the Paris region and add a new route in the destination paste the copied CIDR of the London region's private subnet, then select the target Peering connection and save the changes.



- Now as same as above step add a route in the London region's route table of our Custom VPC.
- In the destination add the CIDR of the Paris region's private subnet of our custom VPC.



Now our Peering connection is established. To verify whether our peering connection is correctly established or not we launch instances in both regions and check it by pinging.

Step 8) *Launching of the Instances in our custom VPC, and*

- *Launch all instances with the rule SSH and All TCMP IPv4.*
- *In Paris Region launch 1 Public instance and another Private instance.*
- *In the London region launch 1 private instance.*
- *After the instances are launched, connect the public instance of the Paris region.*
- *Then Get SSH of the private Instance of the Paris region by using a Bastion Host or Jump Server.*

Step 9) *Now copy the Private IP address of the Private Instance launched in the London Region and ping it by the private instance in the Paris region.*

```
[ec2-user@ip-10-10-8-102 ~]$ ping 20.20.0.81
PING 20.20.0.81 (20.20.0.81) 56(84) bytes of data.
64 bytes from 20.20.0.81: icmp_seq=1 ttl=127 time=10.3 ms
64 bytes from 20.20.0.81: icmp_seq=2 ttl=127 time=10.2 ms
64 bytes from 20.20.0.81: icmp_seq=3 ttl=127 time=10.2 ms
64 bytes from 20.20.0.81: icmp_seq=4 ttl=127 time=10.3 ms
64 bytes from 20.20.0.81: icmp_seq=5 ttl=127 time=10.2 ms
64 bytes from 20.20.0.81: icmp_seq=6 ttl=127 time=10.3 ms
^C
--- 20.20.0.81 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5006ms
rtt min/avg/max/mdev = 10.181/10.258/10.344/0.052 ms
[ec2-user@ip-10-10-8-102 ~]$
```

Now here we can see it is pinging, which means our peering connection is successfully established.