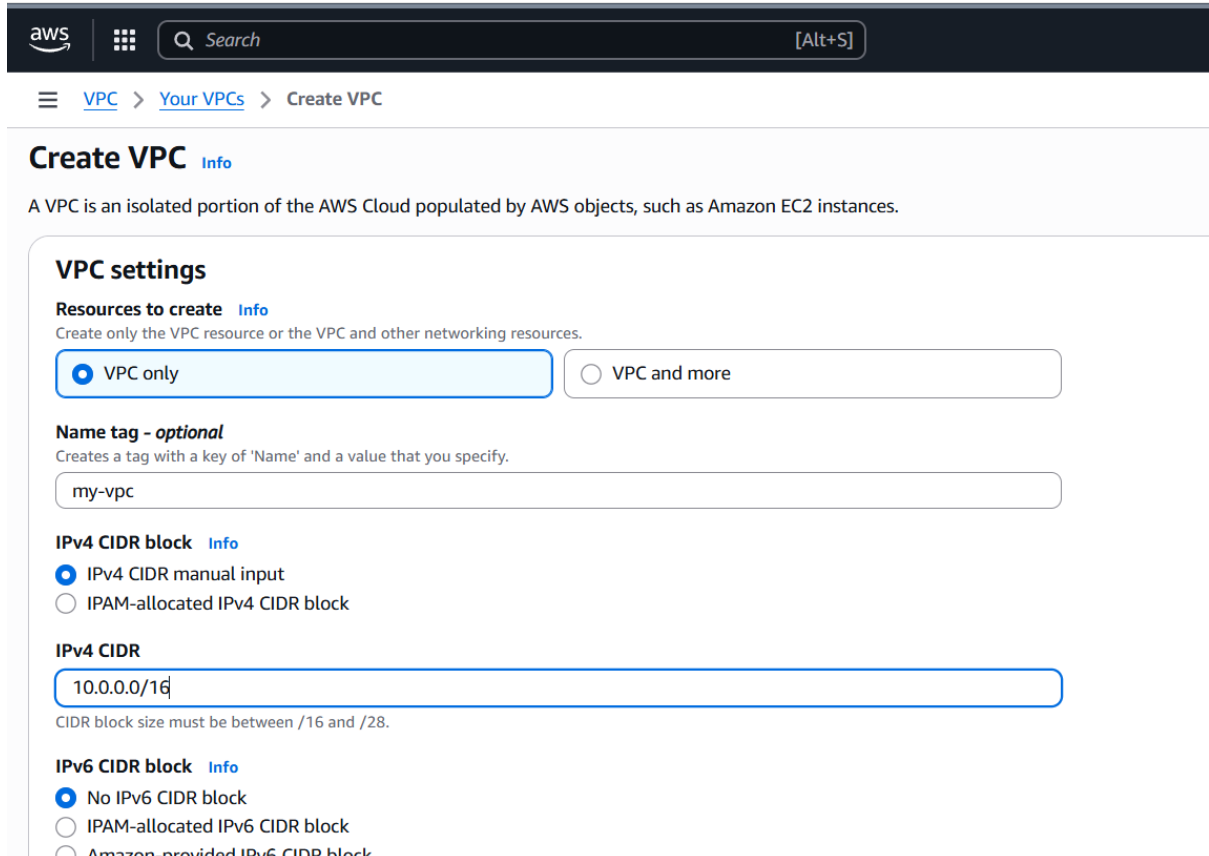


1. Create one VPC, with 1 public subnet and 1 private subnet.

Create a vpc with the name my-vpc.



The screenshot shows the AWS Management Console interface for creating a new VPC. The top navigation bar includes the AWS logo, a search bar, and the breadcrumb path: VPC > Your VPCs > Create VPC. The main heading is 'Create VPC' with an 'Info' link. Below this, a descriptive sentence states: 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.'

The 'VPC settings' section contains the following configuration options:

- Resources to create** (with an 'Info' link): A description states 'Create only the VPC resource or the VPC and other networking resources.' There are two radio button options: 'VPC only' (which is selected) and 'VPC and more'.
- Name tag - optional** (with an 'Info' link): A description states 'Creates a tag with a key of 'Name' and a value that you specify.' The text input field contains 'my-vpc'.
- IPv4 CIDR block** (with an 'Info' link): Two radio button options are present: 'IPv4 CIDR manual input' (selected) and 'IPAM-allocated IPv4 CIDR block'.
- IPv4 CIDR**: A text input field contains '10.0.0.0/16'. A note below the field states: 'CIDR block size must be between /16 and /28.'
- IPv6 CIDR block** (with an 'Info' link): Three radio button options are present: 'No IPv6 CIDR block' (selected), 'IPAM-allocated IPv6 CIDR block', and 'Amazon-provided IPv6 CIDR block'.

Create a public subnet .

aws

Search

[Alt+S]

VPC > Subnets > Create subnet

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.

my-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.

Europe (Stockholm) / eun1-az1 (eu-north-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

Create a private subnet.

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.

my-private-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.

No preference

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

10.0.16.0/20

4,096 IPs

2. Enable VPC peering for cross-region.

Click on peering connections and give the details and give cross-region and vpc id of acceptor and create peering.

The screenshot shows the 'Create peering connection' page in the AWS Management Console. The breadcrumb navigation is 'VPC > Peering connections > Create peering connection'. The page title is 'Create peering connection' with a subtitle explaining that a VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them privately. The 'Peering connection settings' section includes a 'Name' field with the value 'ohio-peering'. The 'Select a local VPC to peer with' section shows 'VPC ID (Requester)' as 'vpc-0721df79a41acf611 (default)'. Below this, a table lists 'VPC CIDRs for vpc-0721df79a41acf611 (default)' with one entry: CIDR '172.31.0.0/16', Status 'Associated', and Status reason '-'. The 'Select another VPC to peer with' section has radio buttons for 'My account' (selected) and 'Another account'.

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.

ohio-peering

Select a local VPC to peer with

VPC ID (Requester)
vpc-0721df79a41acf611 (default)

VPC CIDRs for vpc-0721df79a41acf611 (default)

CIDR	Status	Status reason
172.31.0.0/16	Associated	-

Select another VPC to peer with

Account

☒ My account
☐ Another account

Accept the peering request in the another region.

The screenshot shows the 'Peering connections' page in the AWS Management Console for the 'United States (N. Virginia)' region. The breadcrumb navigation is 'VPC > Peering connections'. The page title is 'Peering connections (1/1) Info'. A table lists one peering connection with ID 'pcx-01459bd03baf45920', status 'Pending acceptance', and requester VPC 'vpc-0721df79a41acf611'. An 'Actions' dropdown menu is open, showing options: 'View details', 'Accept request', 'Reject request', 'Edit DNS settings', 'Manage tags', and 'Delete peering connection'. The 'Details' tab is selected for the connection 'pcx-01459bd03baf45920'. The details section shows 'Requester owner ID' as '72E3E10704EE' and 'VPC Peering connection ARN' as 'arn:aws:ec2:us-east-1:72E3E10704EE:vpc-peering-co-685e / my-685e'.

Peering connections (1/1) Info

Find peering connections by attribute or tag

Name	Peering connection ID	Status	Requester VPC
-	pcx-01459bd03baf45920	Pending acceptance	vpc-0721df79a41acf611

Actions

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

pcx-01459bd03baf45920

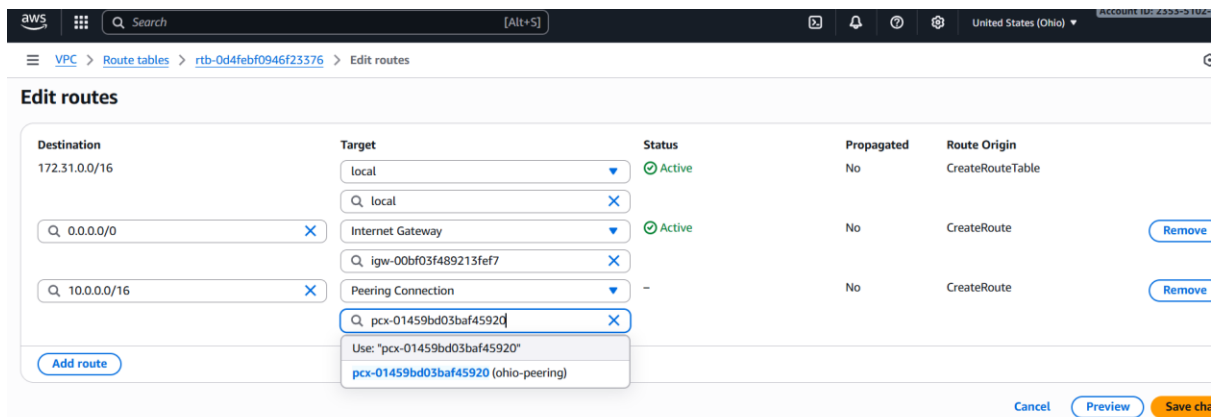
Details

Requester owner ID: 72E3E10704EE

Accepter owner ID: 72E3E10704EE

VPC Peering connection ARN: arn:aws:ec2:us-east-1:72E3E10704EE:vpc-peering-co-685e / my-685e

Go to routetable and edit with another region's cidr range.do same process in the another region also.



Connect with public ip and ping another region's private ip to check the connection.

```
[root@ip-172-31-32-120 ~]# ping 10.0.15.89
PING 10.0.15.89 (10.0.15.89) 56(84) bytes of data.
64 bytes from 10.0.15.89: icmp_seq=416 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=417 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=418 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=419 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=420 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=421 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=422 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=423 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=424 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=425 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=426 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=427 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=428 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=429 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=430 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=431 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=432 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=433 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=434 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=435 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=436 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=437 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=438 ttl=127 time=13.7 ms
64 bytes from 10.0.15.89: icmp_seq=439 ttl=127 time=13.7 ms
```

3. Enable VPC peering for cross-account (you can collaborate with your friend to do this task).

Create peering connection and give details of another region of different account.

The screenshot shows the AWS Management Console interface for creating a VPC peering connection. The breadcrumb navigation at the top indicates the path: VPC > Peering connections > Create peering connection. The main heading is 'Create peering connection', followed by a descriptive sentence and a link to 'Info'. The 'Peering connection settings' section includes a 'Name' field with the value 'devendra-peering'. Below this is a section to 'Select a local VPC to peer with', showing a dropdown for 'VPC ID (Requester)' with 'vpc-0721df79a41acf611 (default)' selected. A table below lists 'VPC CIDRs for vpc-0721df79a41acf611 (default)' with one entry: CIDR '172.31.0.0/16', Status 'Associated' (with a green checkmark), and Status reason '-'. The 'Select another VPC to peer with' section has radio buttons for 'Account', with 'Another account' selected.

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.
devendra-peering

Select a local VPC to peer with

VPC ID (Requester)
vpc-0721df79a41acf611 (default)

VPC CIDRs for vpc-0721df79a41acf611 (default)

CIDR	Status	Status reason
172.31.0.0/16	✔ Associated	-

Select another VPC to peer with

Account
☐ My account
☒ Another account

Accept the request from receivers account.

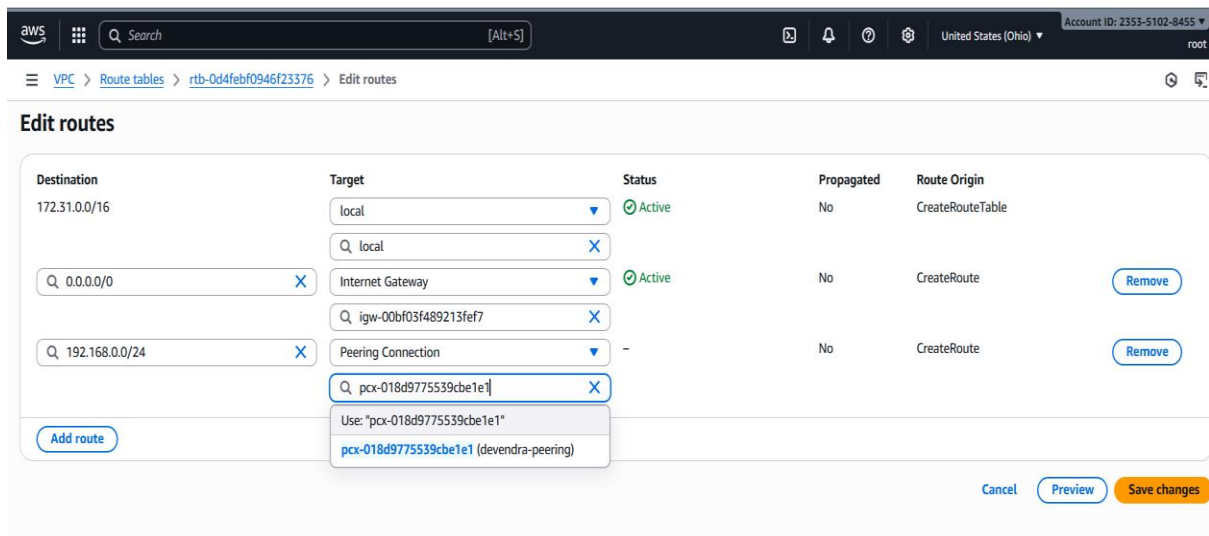
The screenshot shows a green notification banner at the top of the console stating: 'Your VPC peering connection (pcx-018d9775539cbe1e1) has been established. To send and receive traffic across this VPC peering connection, you must add a route to the peered VPC in one or more of your VPC route tables.' A button 'Modify my route tables now' is present. Below the notification, the 'Peering connections (2)' section is visible, including a search bar and a 'Create peering connection' button.

✔ Your VPC peering connection (pcx-018d9775539cbe1e1) has been established.
To send and receive traffic across this VPC peering connection, you must add a route to the peered VPC in one or more of your VPC route tables. [Modify my route tables now](#)

Peering connections (2) [Info](#) [Actions](#) [Create peering connection](#)

Find peering connections by attribute or tag

Add routes in sender's account and receiver's account.

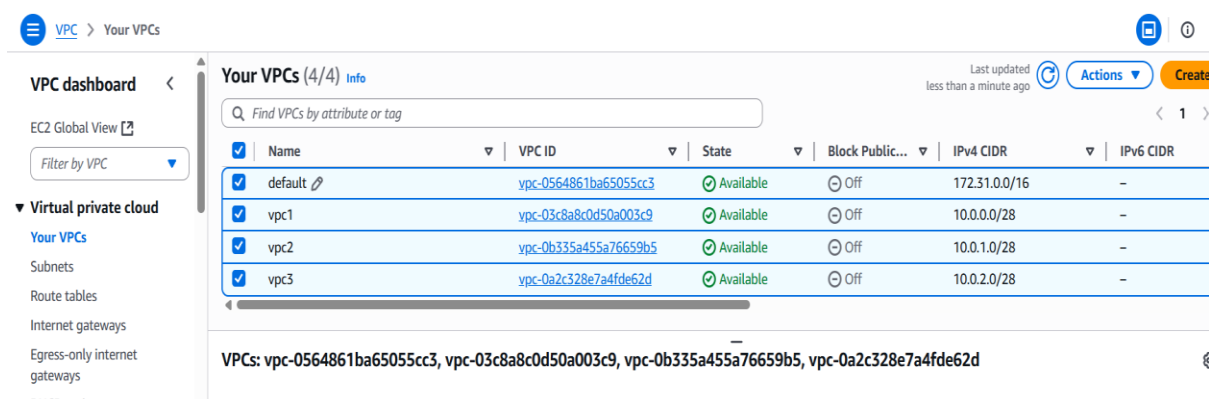


login with your public ip and ping<reciver's private I'd> to check the connectivity.

```
[root@ip-172-31-13-26 ~]# ping 10.0.2.10
PING 10.0.2.10 (10.0.2.10) 56(84) bytes of data.
64 bytes from 10.0.2.10: icmp_seq=1 ttl=126 time=2.32 ms
64 bytes from 10.0.2.10: icmp_seq=2 ttl=126 time=1.22 ms
64 bytes from 10.0.2.10: icmp_seq=3 ttl=126 time=1.13 ms
64 bytes from 10.0.2.10: icmp_seq=4 ttl=126 time=1.14 ms
64 bytes from 10.0.2.10: icmp_seq=5 ttl=126 time=1.13 ms
64 bytes from 10.0.2.10: icmp_seq=6 ttl=126 time=1.15 ms
64 bytes from 10.0.2.10: icmp_seq=7 ttl=126 time=1.13 ms
64 bytes from 10.0.2.10: icmp_seq=8 ttl=126 time=1.12 ms
```

4. Set up a VPC Transit Gateway.

Create 4 vpc's.



Create subnets to the vpc's.

You have successfully created 1 subnet: subnet-06a1c6459d5b9ae38

Subnets (5) [Info](#)

Last updated less than a minute ago [Actions](#) [Create subnet](#)

Find subnets by attribute or tag

<input type="checkbox"/>	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
<input type="checkbox"/>	pub-subnet-default	subnet-02560f8bc88fe708a	Available	vpc-0564861ba65055cc3 defa...	Off	172.31.0.0
<input type="checkbox"/>	pri-subnet-default	subnet-0d4ea6ce22b4e7e6c	Available	vpc-0564861ba65055cc3 defa...	Off	172.31.16.
<input type="checkbox"/>	vpc1-subnet	subnet-0d95feb509064af46	Available	vpc-03c8a8c0d50a003c9 vpc1	Off	10.0.0.0/21
<input type="checkbox"/>	vpc2-subnet	subnet-039d3584bd67b82db	Available	vpc-0b335a455a76659b5 vpc2	Off	10.0.1.0/21
<input type="checkbox"/>	vpc3-subnet	subnet-06a1c6459d5b9ae38	Available	vpc-0a2c328e7a4fde62d vpc3	Off	10.0.2.0/21

Create a transit gateway.

VPC > Transit gateways

connections
Client VPN endpoints

AWS Verified Access
Verified Access instances
Verified Access trust providers
Verified Access groups
Verified Access endpoints

Transit gateways
[Transit gateways](#)

You successfully created tgw-0f4b076d1cff84df5 / my-transitgateway.

You can visualize and monitor your Transit Gateway(s) from the [AWS Network Manager](#). Register your Transit Gateway by creating a [global network](#) to get started.

Transit gateways (1/1) [Info](#)

Find transit gateway by attribute or tag

<input checked="" type="checkbox"/>	Name	Transit gateway ID	State
<input checked="" type="checkbox"/>	my-transitgateway	tgw-0f4b076d1cff84df5	Pending

Create 4 transit gateway attachments and attach to 4 vpc's.

VPC > Transit gateway attachments

connections
Client VPN endpoints

AWS Verified Access
Verified Access instances
Verified Access trust providers
Verified Access groups
Verified Access endpoints

Transit gateways
[Transit gateways](#)
[Transit gateway attachments](#)
Transit gateway policy tables
Transit gateway route tables

You successfully created VPC attachment tgw-attach-0be6fe8d40f1c0282 / vpc3-tga.

Transit gateway attachments (4/4) [Info](#)

Find transit gateway attachment by attribute or tag

<input checked="" type="checkbox"/>	Name	Transit gateway attachment ID	Transit gateway ID	State	Resource type	Resource ID
<input checked="" type="checkbox"/>	vpc1-tga	tgw-attach-021713307685d0ea0	tgw-0f4b076d1cff84df5	Pending	VPC	vpc-03c8a8c0d5
<input checked="" type="checkbox"/>	vpc2-tga	tgw-attach-05a10c5c0760a1751	tgw-0f4b076d1cff84df5	Pending	VPC	vpc-0b335a455e
<input checked="" type="checkbox"/>	vpc3-tga	tgw-attach-0be6fe8d40f1c0282	tgw-0f4b076d1cff84df5	Pending	VPC	vpc-0a2c328e7a
<input checked="" type="checkbox"/>	default-tga	tgw-attach-0e1a25d309a54ff95	tgw-0f4b076d1cff84df5	Available	VPC	vpc-0564861ba6

Transit gateway attachment IDs

[tgw-attach-021713307685d0ea0](#), [tgw-attach-05a10c5c0760a1751](#), [tgw-attach-0be6fe8d40f1c0282](#), [tgw-attach-0e1a25d309a54ff95](#)

Create 4 instances for 4 vpc's.

Instances (4/6) Info

Find Instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
pri-ec2	i-05f85f900276ae46e	Terminated	t3.micro	–	View alarms +	us-west-1a	–
vpc3-ec2	i-0aa3f349b0aa2fb9a	Running	t3.micro	Initializing	View alarms +	us-west-1a	–
vpc1-ec2	i-012697a6b4e68a3cc	Running	t3.micro	Initializing	View alarms +	us-west-1a	–
pub-ec2	i-0e782b07ee52add85	Terminated	t3.micro	–	View alarms +	us-west-1c	–
default-ec2	i-04c06ab5dbc089490	Running	t3.micro	3/3 checks passed	View alarms +	us-west-1c	ec2-5
vpc2-ec2	i-034198f5fabdefc00	Running	t3.micro	Initializing	View alarms +	us-west-1c	–

4 instances selected

Monitoring

Configure CloudWatch agent

Go to route table and edit the routes give permissions for all vpc's.

VPC dashboard

Route tables

rtb-0c3340b3f557c1b0d

Updated routes for rtb-0c3340b3f557c1b0d / default-rt successfully

Details

Route table ID: rtb-0c3340b3f557c1b0d

Main: Yes

Explicit subnet associations: subnet-02560f8bc88fe708a / pub-subnet-default

Edge associations: –

VPC: vpc-0564861ba65055cc3 | default

Owner ID: 235351028455

Routes (5)

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-021348c6279231955	Active	No	Create Route
10.0.0.0/28	tgw-0f4b076d1cff84df5	Active	No	Create Route
10.0.1.0/28	tgw-0f4b076d1cff84df5	Active	No	Create Route
10.0.2.0/28	tgw-0f4b076d1cff84df5	Active	No	Create Route
172.31.0.0/16	local	Active	No	Create Route Table

We need to create internet gateways for 4 vpc's.

Internet gateways (1/2) Info

Find internet gateways by attribute or tag

Name	Internet gateway ID	State	VPC ID
default-internetgateway	igw-021348c6279231955	Attached	vpc-0564861ba65055cc3 default
vpc2-internetgateway	igw-04effb92417c5492c	Attached	vpc-0b335a455a76659b5 vpc2

igw-021348c6279231955

Login with public ip address and ping any instance private ip address.

```
[root@ip-172-31-13-26 ~]# ping 10.0.2.10
PING 10.0.2.10 (10.0.2.10) 56(84) bytes of data.
64 bytes from 10.0.2.10: icmp_seq=1 ttl=126 time=2.32 ms
64 bytes from 10.0.2.10: icmp_seq=2 ttl=126 time=1.22 ms
64 bytes from 10.0.2.10: icmp_seq=3 ttl=126 time=1.13 ms
64 bytes from 10.0.2.10: icmp_seq=4 ttl=126 time=1.14 ms
64 bytes from 10.0.2.10: icmp_seq=5 ttl=126 time=1.13 ms
64 bytes from 10.0.2.10: icmp_seq=6 ttl=126 time=1.15 ms
64 bytes from 10.0.2.10: icmp_seq=7 ttl=126 time=1.13 ms
64 bytes from 10.0.2.10: icmp_seq=8 ttl=126 time=1.12 ms
^C
--- 10.0.2.10 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7009ms
rtt min/avg/max/mdev = 1.121/1.292/2.321/0.389 ms
[root@ip-172-31-13-26 ~]# ping 10.0.0.6
PING 10.0.0.6 (10.0.0.6) 56(84) bytes of data.
64 bytes from 10.0.0.6: icmp_seq=1 ttl=126 time=1.93 ms
64 bytes from 10.0.0.6: icmp_seq=2 ttl=126 time=1.08 ms
64 bytes from 10.0.0.6: icmp_seq=3 ttl=126 time=1.08 ms
64 bytes from 10.0.0.6: icmp_seq=4 ttl=126 time=1.09 ms
64 bytes from 10.0.0.6: icmp_seq=5 ttl=126 time=1.05 ms
64 bytes from 10.0.0.6: icmp_seq=6 ttl=126 time=1.07 ms
^C
--- 10.0.0.6 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5007ms
rtt min/avg/max/mdev = 1.048/1.217/1.931/0.319 ms
```

5. Set up a VPC Endpoint.

Open ec2 instance with public ip address.

```

MUJJU SK@DESKTOP-LU541U4 MINGW64 ~/Downloads
$ ssh -i california.pem ec2-user@54.183.26.154
The authenticity of host '54.183.26.154 (54.183.26.154)' can't be established.
ED25519 key fingerprint is SHA256:zLM1ROVXJkpgA5yITcCKbmqx5QeERCiEu1D3ZkvZk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.183.26.154' (ED25519) to the list of known hosts.

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

[ec2-user@ip-172-31-6-12 ~]$ sudo su-
sudo: su-: command not found
[ec2-user@ip-172-31-6-12 ~]$ sudo su -
[root@ip-172-31-6-12 ~]# aws --version
aws-cli/2.27.57 Python/3.9.23 Linux/6.1.150-174.273.amzn2023.x86_64 source/x86_64.amzn.2023
[root@ip-172-31-6-12 ~]#

```

Create an end point for public instance.

Create endpoint [Info](#)

Create the type of VPC endpoint that supports the service, service network or resource to which you want to connect.

Endpoint settings

Specify a name and select the type of endpoint.

Name tag - *optional*

Creates a tag with a key of 'Name' and a value that you specify. Tags help you find and manage your endpoint.

s3-endpoint

Type [Info](#)

Select a category

☒ **AWS services**
Connect to services provided by Amazon with an Interface endpoint, or a Gateway endpoint

☐ **EC2 Instance Connect Endpoint**
An elastic network interface that allows you to connect to resources in a private subnet

☐ **PrivateLink Ready partner service**
Connect to SaaS services which have an AWS PrivateLink Interface endpoint. Uses AWS PrivateLink

☐ **Resources**
Connect to resources like Amazon Relational Database Service. Uses AWS PrivateLink Resource endpoint. Uses AWS PrivateLink

Services (1/2)

🔍 Search

Service Name = com.amazonaws.us-west-1.s3 ✕

[Clear filters](#)

	Service Name	Owner	Type
<input type="radio"/>	com.amazonaws.us-west-1.s3	amazon	Interface
<input checked="" type="radio"/>	com.amazonaws.us-west-1.s3	amazon	Gateway

Network settings

Select the VPC in which to create the endpoint

VPC

Create the VPC endpoint in the VPC in the same AWS Region from which you will access a resource.

vpc-0564861ba65055cc3

Route tables (1/2) [Info](#)

🔍 Search

[VPC](#) / [Endpoints](#) / Create endpoint

vpc-0564861ba65055cc3

route tables (1/2) [Info](#)

Search

<input type="checkbox"/>	Name	Route Table ID	Main
<input type="checkbox"/>	pub-route-table	rtb-0c3340b3f557c1b0d (pub-route-ta...	Yes
<input checked="" type="checkbox"/>	pri-routetable	rtb-0b334e4702f6e460c (pri-routetable)	No

When you use an endpoint, the source IP addresses from your instances in your affected subnets for accessing t from your affected subnets to the AWS service that use public IP addresses may be dropped. Ensure that you d

rtb-0b334e4702f6e460c X

Policy [Info](#)

PC endpoint policy controls access to the service.

Full access

Allow access by any user or service within the VPC using credentials from any Amazon Web Services accounts to any resources i

We can see endpoint is connected to bucket.

```
connecting credentials
[root@ip-172-31-6-12 .aws]# cd ~
[root@ip-172-31-6-12 ~]# aws s3 ls
2025-09-23 12:39:35 my-s3bucket0123
[root@ip-172-31-6-12 ~]#
```