

NAT GATEWAY LAB

Create a VPC

[VPC](#) > [Your VPCs](#) > [Create VPC](#)

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

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.

IPv4 CIDR block Info

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

IPv4 CIDR

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

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	
<input type="text" value="Name"/>	<input type="text" value="vpc1"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

Create Subnets one for public and one for private

- Public

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.

vpc-0ae4ac7e18b0bb366 (vpc1) ▼

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.

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.

No preference ▼

IPv4 CIDR block Info

10.0.0.0/24 ✕

▼ Tags - *optional*

Key	Value - <i>optional</i>	
✕ Name ✕	✕ Public_subnet ✕	Remove

Add new tag

You can add 49 more tags.

Remove

Add new subnet

- Private

Create subnet [Info](#)

VPC

VPC ID
Create subnets in this VPC.

vpc-0ae4ac7e18b0bb366 (vpc1) ▼

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.

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 CIDR block [Info](#)

10.0.1.0/24 ✕

▼ **Tags - optional**

Key	Value - optional	
✕ Name	✕ Private_subnet	Remove

Add new tag

You can add 49 more tags.

Remove

Add new subnet

Create Internet Gateway and attach it to the VPC you have created

VPC > Internet gateways > Create internet gateway

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.

No tags associated with the resource.

[Add new tag](#)

You can add 50 more tags.

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

Select you Internet Gateway and click actions there you will find attach option, after clicking on it you can attach you IG to a VPC

VPC > Internet gateways > Attach to VPC (igw-00a29fd0f0d09ba8c)

Attach to VPC (igw-00a29fd0f0d09ba8c) [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.

vpc-0ae4ac7e18b0bb366 - vpc1

▶ AWS Command Line Interface command

[Cancel](#) [Attach internet gateway](#)

You can see here your IG is attached to the VPC you have created

VPC > Internet gateways > igw-00a29fd0f0d09ba8c

igw-00a29fd0f0d09ba8c / vpc1-ig

Actions ▾

Details Info

Internet gateway ID igw-00a29fd0f0d09ba8c	State Attached	VPC ID vpc-0ae4ac7e18b0bb366 vpc1	Owner 436117849909
--	-------------------	--	-----------------------

Tags Manage tags

Search tags

< 1 > ⚙

Key	Value
Name	vpc1-ig

Go to route table and add Internet gateway to your main route table:

VPC > Route tables > rtb-0068b19517428a96b > Edit routes

Edit routes

Edit routes

Destination	Target	Status
10.0.0.0/16	local	Active
Propagated		
No		

Edit routes

Destination	Target	Status
0.0.0.0/0	igw-00a29fd0f0d09ba8c	-
Propagated		
No		

Remove

Add route

Cancel Preview Save changes

Create another Route table for the VPC with no IG attached to it

VPC > Route tables > rtb-09ad37fce832d8184

rtb-09ad37fce832d8184 / vpc1-rt(Private)

Actions

DetailsInfo

Route table ID

rtb-09ad37fce832d8184

VPC

vpc-0ae4ac7e18b0bb366 | vpc1

Main

No

Owner ID

436117849909

Explicit subnet associations

-

Edge associations

-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (1)

Edit routes

Filter routes

Both

< 1 >

Destination

Target

Status

Propagated

10.0.0.0/16

local

Active

No

Associate Subnets to the route table by edit subnet association:

- Here public subnet will be attached to the route table with IG such that main route table of vpc1
- And private subnet to the route table with no IG association

Below image shows the public subnet being associated to the route table with IG (click on the section of explicit subnet association to verify which subnet is attached to the route table)

VPC > Route tables > rtb-0068b19517428a96b

rtb-0068b19517428a96b / vpc1-rt(Public)

Actions ▾

Details Info

Route table ID rtb-0068b19517428a96b	Main Yes	Explicit subnet associations subnet-069b102ff3b5117be / public_subnet	Edge associations -
VPC vpc-0ae4ac7e18b0bb366 vpc1	Owner ID 436117849909		

Routes Subnet associations Edge associations Route propagation Tags

Routes (2) Edit routes

Filter routes Both < 1 > ⚙

Destination ▾	Target ▾	Status ▾	Propagated ▾
0.0.0.0/0	igw-00a29fd0f0d09ba8c	Active	No
10.0.0.0/16	local	Active	No

Similarly attach the private subnet to the route table with no IG associated to it.

VPC > Route tables > rtb-09ad37fce832d8184

rtb-09ad37fce832d8184 / vpc1-rt(Private)

Actions

DetailsInfo

Route table ID
rtb-09ad37fce832d8184

VPC
vpc-0ae4ac7e18b0bb366 | vpc1

Main
No

Owner ID
436117849909

Explicit subnet associations
subnet-0297148b8b63c1e77 / private_subnet

Edge associations
-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (1)

Edit routes

Filter routes

Both

< 1 >

Destination

Target

Status

Propagated

10.0.0.0/16

local

Active

No

Below image shows your private subnet associated with the route table with no IG associated to it

VPC > Route tables > rtb-09ad37fce832d8184

rtb-09ad37fce832d8184 / vpc1-rt(Private)

Actions

DetailsInfo

Route table ID
rtb-09ad37fce832d8184

VPC
vpc-0ae4ac7e18b0bb366 | vpc1

Main
No

Owner ID
436117849909

Explicit subnet associations
subnet-0297148b8b63c1e77 / private_subnet

Edge associations
-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (1)

Edit routes

Filter routes

Both

< 1 >

Destination

Target

Status

Propagated

10.0.0.0/16

local

Active

No

Create a NAT Gateway:

- ⚠ NAT Gateway is created in a public subnet, hence we select public subnet in the Subnet option shown in figure.

[VPC](#) > [NAT gateways](#) > Create NAT gateway

Create NAT gateway [Info](#)

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

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

The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.

subnet-069b102ff3b5117be (public_subnet) ▼

Connectivity type
Select a connectivity type for the NAT gateway.

☒ Public
☐ Private

Elastic IP allocation ID [Info](#)
Assign an Elastic IP address to the NAT gateway.

eipalloc-066959e8c89165709 ▼

Allocate Elastic IP

► **Additional settings** [Info](#)

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	
<input type="text" value="Name"/> X	<input type="text" value="vpc1-ng"/> X	Remove

Add new tag

You can add 49 more tags.

Cancel **Create NAT gateway**

Edit Route table attached to private subnet

- Add NAT gateway route to the route table

VPC > Route tables > rtb-09ad37fce832d8184

rtb-09ad37fce832d8184 / vpc1-rt(Private)

Actions ▾

Details Info

Route table ID rtb-09ad37fce832d8184	Main No	Explicit subnet associations subnet-0297148b8b63c1e77 / private_subnet	Edge associations -
VPC vpc-0ae4ac7e18b0bb366 vpc1	Owner ID 436117849909		

Routes Subnet associations Edge associations Route propagation Tags

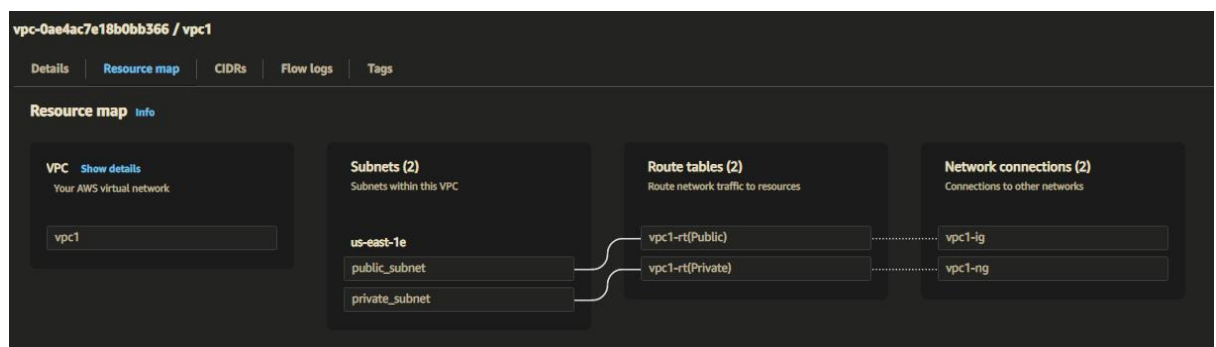
Routes (2) Edit routes

Filter routes Both ▾ < 1 > ⚙

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0989a35fbdcff0c96	Active	No
10.0.0.0/16	local	Active	No

Check the resource map you will see the routes you have created

- Below you can see how you have associated your public subnet to the IG and private subnet with the NAT gateway



- Edit network settings while creating EC2 instances, select public subnet and enable auto-assign public IP to make ec2 publicly accessible and select private subnet for private ec2 and disable auto-assign public IP. Create common security group vpc1-sg for both instances

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-0ae4ac7e18b0bb366 (vpc1)
10.0.0.0/16

↻

Subnet [Info](#)

subnet-069b102ff3b5117be
VPC: vpc-0ae4ac7e18b0bb366 Owner: 436117849909
Availability Zone: us-east-1e IP addresses available: 250 CIDR: 10.0.0.0/24

public_subnet
↻

↻ Create new subnet [↗](#)

Auto-assign public IP [Info](#)

Enable

▼

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

vpc1-sg

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-2 created 2023-02-19T13:42:10.581Z

Inbound security groups rules

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

Remove

Type [Info](#)

Protocol [Info](#)

Port range [Info](#)

ssh
▼

TCP

22

Source type [Info](#)

Source [Info](#)

Description - optional [Info](#)

Anywhere
▼

🔍 Add CIDR, prefix list or security

0.0.0.0/0 ✕

e.g. SSH for admin desktop

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

Add security group rule

Edit the security group of the instances:

- Add following inbound rules to your Security groups as shown in the image below.

EC2 > Security Groups > sg-01e2fe4241661c69b - vpc1-sg

sg-01e2fe4241661c69b - vpc1-sg Actions ▾

Details

Security group name vpc1-sg	Security group ID sg-01e2fe4241661c69b	Description launch-wizard-2 created 2023-02-19T12:55:00.334Z	VPC ID vpc-0ae4ac7e18b0bb366
Owner 436117849909	Inbound rules count 4 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules | Outbound rules | Tags

Inbound rules (4)

Filter security group rules < 1 > ⚙

	Name ▾	Security group r... ▾	IP versi... ▾	Type ▾	Protocol ▾	Port ra... ▾	Source ▾
■	-	sgr-005ee7766def...	IPv4	All ICMP - ...	ICMP	All	0.0.0.0/0
■	-	sgr-0b95690c7d11...	IPv4	SSH	TCP	22	0.0.0.0/0
■	-	sgr-0efcc0fd5329af...	IPv4	HTTPS	TCP	443	0.0.0.0/0
■	-	sgr-0dd250ad4d93...	IPv4	HTTP	TCP	80	0.0.0.0/0

Open Public instance using putty

```
root@ip-10-0-0-151:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
  _ |  ( _ | _ )  
  _ |  /      Amazon Linux 2 AMI  
  _ | \ _ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-10-0-0-151 ~]$ sudo su -  
[root@ip-10-0-0-151 ~]#
```

- enter “sudo su –” command to switch to root user. Now you have to access your private subnet from the public subnet for that go to private Ec2 instance we have created and click on connect and then click on SSH client, you will see the following procedure there

EC2 > Instances > i-049c01bb6c5855730 > Connect to instance

Connect to instance Info

Connect to your instance i-049c01bb6c5855730 (private-m) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID
i-049c01bb6c5855730 (private-m)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is Dsp.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 Dsp.pem
4. Connect to your instance using its Private IP:

✔ Command copied

```
ssh -i "Dsp.pem" ec2-user@10.0.139
```

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

Cancel

Execute following in your linux CLI :

- “vi Dsp.pem”
- open your .pem file of your key assigned to the instance in Notepad and copy the key
- In CLI press “I” you will enter into insert mode and now paste the copied key.
- Now copy the commands from the connect to instance, as shown in the image above.
- After successful connection you will see the following ,you are now into private instance.

```
[root@ip-10-0-0-151 ~]# chmod 400 Dsp.pem
[root@ip-10-0-0-151 ~]# ssh -i "Dsp.pem" ec2-user@10.0.1.39
The authenticity of host '10.0.1.39 (10.0.1.39)' can't be established.
ECDSA key fingerprint is SHA256:1W8UmaAj950MaecPxCw5MDepBEDP4iVP5WsvdqIoaEk.
ECDSA key fingerprint is MD5:18:f1:6b:21:d3:41:2f:36:b7:0a:f7:3d:34:eb:14:de.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.1.39' (ECDSA) to the list of known hosts.

  _ | _ | _ )
  _ | ( _ /   Amazon Linux 2 AMI
  _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-1-39 ~]$ sudo su -
```

- Type “sudo su –” command and switch to root user
- Now use the command “ping 8.8.8.8” to verify whether the private subnet can access internet, if your private subnet is connected to internet you will see the following output

```
[root@ip-10-0-1-39 ~]# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=106 time=1.97 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=106 time=1.46 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=106 time=1.44 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=106 time=1.43 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=106 time=1.43 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=106 time=1.41 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=106 time=1.41 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=106 time=1.43 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=106 time=1.38 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=106 time=1.49 ms
```

- This means NAT gateway is working and now you can access internet from your private subnet.