

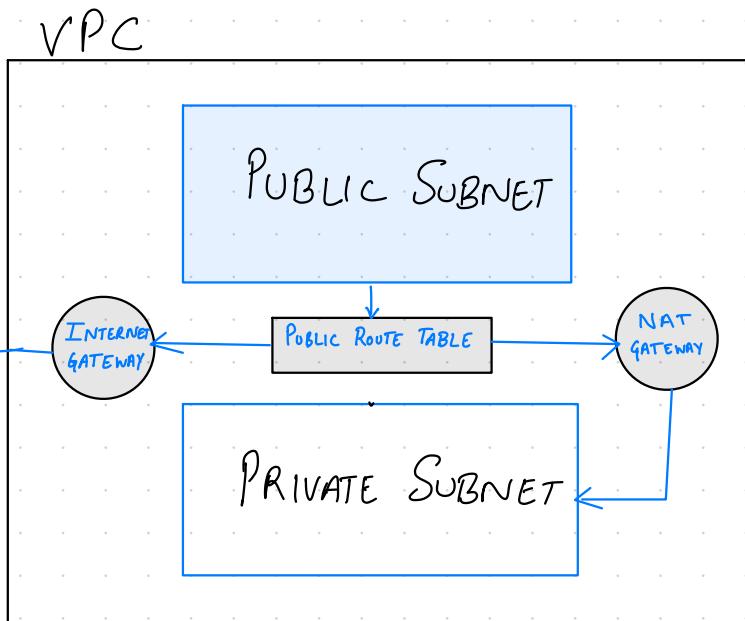
All about AWS VPC Service

By: Chetan Harmon

AWS VPC & VPC Peering

AWS CLOUD (Any Region)

IPUBLIC ACCESS



VPC → Virtual Private cloud

- Isolated environment to set up your application

SUBNET → A range of IP addresses in your VPC.

Route table → A set of rules called routes, that are used to determine where network traffic is directed

Internet gateway → A gateway that you attach to your VPC to enable communication

between resources
in your vpc and
the internet.

CIDR → classes inter
block Domain Routing.

An internet
protocol address
allocation and route
aggregation methodology

Default VPC Components:-

- Create a VPC with a size /16 IPv4 CIDR block 172.0.0.1/16. This provides upto 65,536 private IPv4 addresses.
- Create a size of /20 Default subnet in each availability zone. This provide upto 4,096 address per subnet. A few of

which are reserved for
our use.

- Create an Internet
Gateway and connect
it to your VPC.
- add a route to main
route table that points
all traffic ($0.0.0.0/0$)
to the internet
gateway

o Create a security group
and associate it with
your VPC.

Steps to create VPC:

1. Select my particular region. In this case we will go with Ohio region

The screenshot shows the AWS VPC Console dashboard for the 'us-east-2' region. At the top, there are buttons for 'Create VPC' and 'Launch EC2 Instances'. Below this, a section titled 'Resources by Region' displays various Amazon VPC resources across the 'US East (Ohio)' region. These include:

- VPCs: 1 instance
- NAT Gateways: 0 instances
- Subnets: 3 instances
- VPC Peering Connections: 0 instances
- Route Tables: 0 instances
- Network ACLs: 1 instance
- Internet Gateways: 1 instance
- Security Groups: 1 instance
- Egress-only Internet Gateways: 0 instances
- Customer Gateways: 0 instances

On the right side of the dashboard, there are sections for 'Service Health', 'Settings', 'Additional Information', and 'AWS Network Manager'. The 'Service Health' section includes a link to 'View complete service health details'. The 'Settings' section has links for 'Zones' and 'Console Experiments'. The 'Additional Information' section links to 'VPC Documentation', 'All VPC Resources', 'FAQ', and 'Report an Issue'. The 'AWS Network Manager' section provides a brief description of its features and a link to 'Get started with Network Manager'. The bottom of the page includes standard AWS navigation links like CloudShell, Feedback, and Help, along with a search bar and the AWS logo.

2. Select your VPCs. There we can see the default VPC. Now hit on create VPC.

- give name to your VPC
- select IPv4 CIDR block
- set the IPv4 CIDR range as 172.0.0.0/16
- and then hit create VPC

The screenshot shows the AWS VPC console interface. At the top, there's a navigation bar with links for EC2, Elastic Kubernetes Service, Elastic Container Registry, S3, VPC, CloudWatch, RDS, DynamoDB, Lambda, Elastic Container Service, CloudFormation, API Gateway, and ElastiCache. Below the navigation bar, a search bar is present. The main area displays a table titled "Your VPCs". The table has two rows of data:

| Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR | DHCP option set | Main ro |
|---------------|-----------------------|-----------|---------------|-----------|------------------------|---------|
| - | vpc-0dbefb9e58aea986d | Available | 172.31.0.0/16 | - | dopt-033b039bb074d5... | rtb-06a |
| Development ✓ | vpc-042161f85c87e07da | Available | 172.0.0.0/16 | ✓ | dopt-033b039bb074d5... | - |

3. Next we will create Subnet. In Subnet create window.
- select your VPC.
 - Set subnet name (Public Subnet)
 - Choose any availability zone, out of three.
 - to set IPv4 CIDR block, go to

<https://www.davidc.net/sites/default/subnets/subnets.html>

On this site I set

the VPC CIDR block

orange and set block

as /16 & hit update

Now divide the

CIDR range in

such that public

subnet attain 256

|P₁|

Visual Subnet Calculator

Enter the network you wish to subnet:

| | |
|--|--|
| Network Address | Mask bits |
| <input type="text" value="172.0.0.0"/> | <input type="text" value="16"/> / Update Reset |

Show columns: Subnet address Netmask Range of addresses Usable IPs Hosts Divide Join

[Click below to split and join subnets](#)

If you wish to save this subnetting for later, bookmark this [hyperlink](#).

| Subnet address | Range of addresses | Useable IPs | Hosts | Divide | Join | | | | |
|----------------|-----------------------------|-----------------------------|-------|--------|------|-----|-----|-----|-----|
| 172.0.0.0/24 | 172.0.0.0 - 172.0.0.255 | 172.0.0.1 - 172.0.0.254 | 254 | Divide | /24 | /19 | /18 | /17 | /16 |
| 172.0.1.0/24 | 172.0.1.0 - 172.0.1.255 | 172.0.1.1 - 172.0.1.254 | 254 | Divide | /24 | /23 | /22 | | |
| 172.0.2.0/23 | 172.0.2.0 - 172.0.3.255 | 172.0.2.1 - 172.0.3.254 | 510 | Divide | /23 | /23 | /22 | | |
| 172.0.4.0/22 | 172.0.4.0 - 172.0.7.255 | 172.0.4.1 - 172.0.7.254 | 1022 | Divide | /22 | /21 | /20 | | |
| 172.0.8.0/21 | 172.0.8.0 - 172.0.15.255 | 172.0.8.1 - 172.0.15.254 | 2046 | Divide | | | | | |
| 172.0.16.0/20 | 172.0.16.0 - 172.0.31.255 | 172.0.16.1 - 172.0.31.254 | 4094 | Divide | | | | | |
| 172.0.32.0/19 | 172.0.32.0 - 172.0.63.255 | 172.0.32.1 - 172.0.63.254 | 8190 | Divide | | | | | |
| 172.0.64.0/18 | 172.0.64.0 - 172.0.127.255 | 172.0.64.1 - 172.0.127.254 | 16382 | Divide | | | | | |
| 172.0.128.0/17 | 172.0.128.0 - 172.0.255.255 | 172.0.128.1 - 172.0.255.254 | 32766 | Divide | | | | | |

hit divide to obtain IP Range.

— Now create the Subnet.

The screenshot shows the AWS Management Console with the VPC service selected. A green banner at the top indicates "You have successfully created 1 subnet: subnet-07ac790a82669ee5f". Below this, the "Subnets (1) Info" section displays a single subnet entry:

| Name | Subnet ID | State | VPC | IPv4 CIDR | IPv6 CIDR |
|---------------|--------------------------|-----------|--------------------------------|--------------|-----------|
| Public subnet | subnet-07ac790a82669ee5f | Available | vpc-042161f85c87e07da Dev... | 172.0.0.0/24 | - |

4. Now create Route table
that will associate with
this subnet. In route table
window select :-

- Give name to route table
- Select your VPC (Development)
- Create route table.

A screenshot of the AWS VPC Route Tables console. A green banner at the top says "Route table rtb-04a23baabf863427e / development_Public_subnet was created successfully." Below it, the route table details are shown: Route table ID (rtb-04a23baabf863427e), Main, No, Owner ID (94056461586), Explicit subnet associations (empty), Edge associations (empty). Under the Routes tab, there is one route entry: Destination (172.0.0.0/16), Target (local), Status (Active), Propagated (No). The browser's address bar shows the URL: us-east-1.console.aws.amazon.com/vpc/routes/RouteTables/route-tables?route-tableId=rtb-04a23baabf863427e.

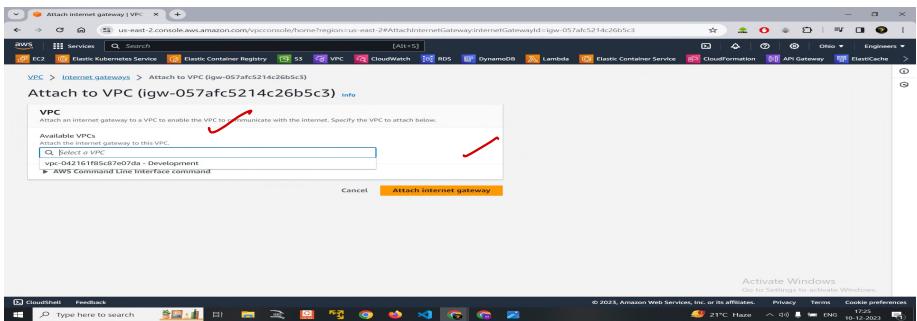
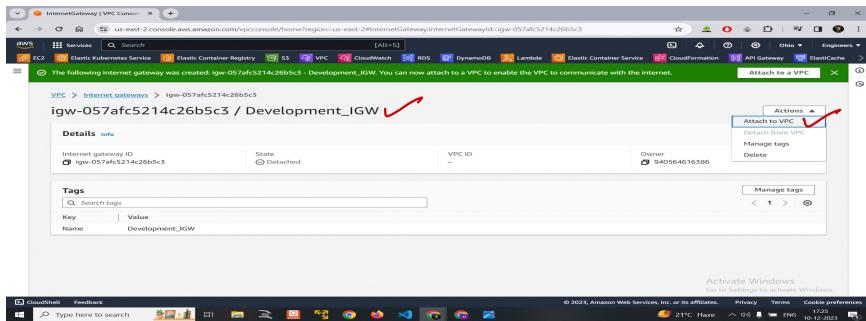
5. Now Associate this to public Subnet.

- Select on Subnet Association & hit edit Subnet association.
Over there select the subnet which we created.

A screenshot of the "Edit Route Table Subnet Associations" dialog. It shows a table with one row: Available subnets (1/1) - Public subnet (selected). The "Selected subnets" section contains "subnet-07ac790a82669ee5f / Public subnet". At the bottom right, there is a yellow "Save associations" button with a red arrow pointing to it. The browser's address bar shows the URL: us-east-1.console.aws.amazon.com/vpc/routes/RouteTables/route-tables?route-tableId=rtb-04a23baabf863427e#edit-subnet-associations.

6. Now our Subnet has Route table. But it isn't public subnet yet. for that we need to attach

internet gateway to this.
So, now create Internet gateway and attach it to VPC.



7. Now we will set route
of all traffic (0.0.0.0/0)
to this internet gateway
that will make the subnet

public

- go to route table
i.e. attached to subnet
in routes choose edit
routes, over there
add route.

The screenshot shows the AWS VPC Edit Routes interface. A route is being added to a route table. The destination is set to 0.0.0.0/0, and the target is set to an Internet Gateway (igw-057afc5214c26b5c3). The route is marked as Active and Propagated. The 'Save changes' button is highlighted with a red checkmark.

8. Now our subnet is public Subnet.

9. To test go to EC2
and over there create
instance inside the
VPC in this public
subnet. Connect to
instance and ping to
8.8.8.8. If the
package receives data
means instance has
internet access.

```
ubuntu@ip-172-0-0-96:~$ 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=43 time=10.7 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=43 time=10.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=43 time=10.7 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=43 time=10.7 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 10.676/10.702/10.740/0.023 ms
ubuntu@ip-172-0-0-96:~$ █
i-07505ac86bed9a544 (Public_Instance)
PublicIPs: 3.15.198.145 PrivateIPs: 172.0.0.96
```

vpc vpc-042161f85c87e07da | us-east-2#LaunchInstances:

Network settings

VPC - required **Info**
vpc-042161f85c87e07da (Development) 172.0.0.0/16

Subnet **Info**
subnet-07ac790a82669ee5f Public subnet
VPC: vpc-042161f85c87e07da Owner: 940564616386 Availability Zone: us-east-2a IP addresses available: 250 CIDR: 172.0.0.0/24

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 **Info**
launch-wizard-2

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 2023-12-10T12:06:56.464Z

Summary

Number of instances **Info**
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more ami-0e83be366243f524a

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which

Cancel Launch instance Activate Windows Go to Settings to activate Windows Review commands

CloudShell Feedback Type here to search 17°C Haze 10-12-2023

10. Till now we have achieved one public Subnet with internet access.

vpcs | VPC Console x Dashboard | EC2 | us-east-2 x +

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#vpcs:

Services Search [Alt+S]

EC2 Elastic Kubernetes Service Elastic Container Registry S3 VPC CloudWatch RDS DynamoDB Lambda Elastic Container Service CloudFormation API Gateway ElastiCache

Your VPCs (1/2) Info

| Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR | DHCP option set | Main route table |
|---|-----------------------|-----------|---------------|-----------|------------------------|------------------|
| - | vpc-0dbefb9e58aea986d | Available | 172.31.0.0/16 | - | dopt-033b039bb074d5... | rtb-06a... |
| <input checked="" type="checkbox"/> Development | vpc-042161f85c87e07da | Available | 172.0.0.0/16 | - | dopt-033b039bb074d5... | - |

vpc-042161f85c87e07da / Development

Details Resource map New CIDRs Flow logs Tags Integrations

Resource map Info

VPC Show details Your AWS virtual network

Development

Subnets (1) Subnets within this VPC

us-east-2a Public subnet

development_Public_subnet rtb-017bb8eaf2e86b6e6

Route tables (2) Route network traffic to resources

Network connections (1) Connections to other networks

Was the resource map helpful today? Give us feedback as often as

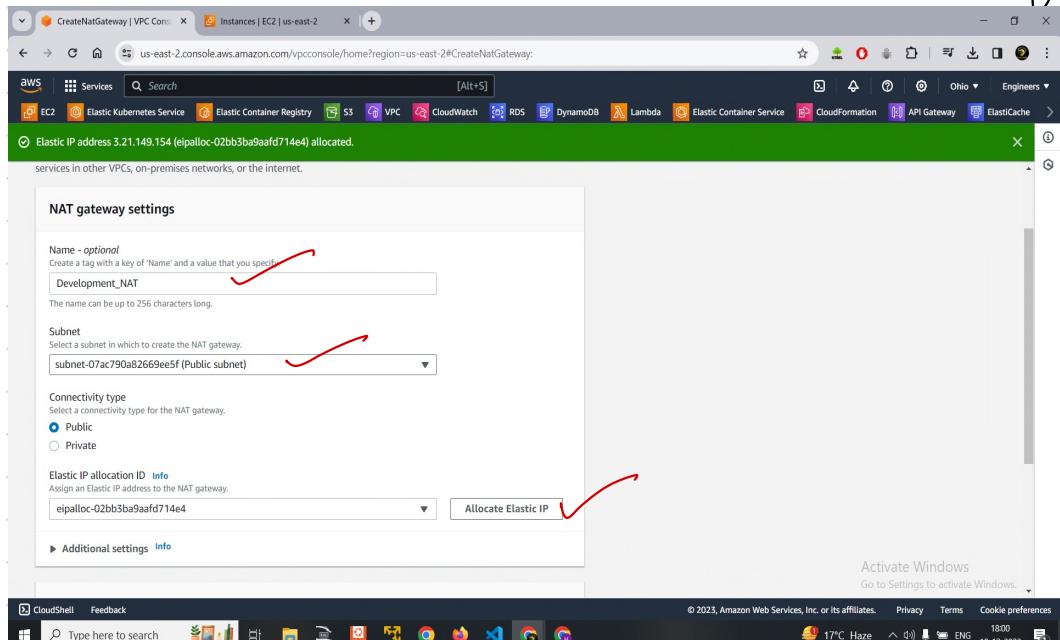
Activate Windows Go to Settings to activate Windows.

CloudShell Feedback Type here to search 21°C Haze ENG 10-12-2023

11. Now we will create one private Subnet similar to public subnet, but it will not be attached to Internet gateway.

12. So, we will create

a NAT Gateway in
public subnet do
allocate elastic IP
to this NAT Gateway



13. Now we will give
IPv4 CIDR range of

all traffic (0.0.0.0/0) to
route table of private
subnet & route that
to NAT gateway.

The screenshot shows the 'Routes' tab of a CloudFormation stack named 'Private_subnet_development'. A red arrow points to the 'Target' column for the first route entry, which is set to 'nat-0317f13fftab20592'. Another red arrow points to the 'Status' column for the same entry, which is labeled 'Active'.

| Destination | Target | Status | Propagated |
|--------------|-----------------------|--------|------------|
| 0.0.0.0/0 | nat-0317f13fftab20592 | Active | No |
| 172.0.0.0/16 | local | Active | No |

14. Now create instance
in the private subnet
and don't give

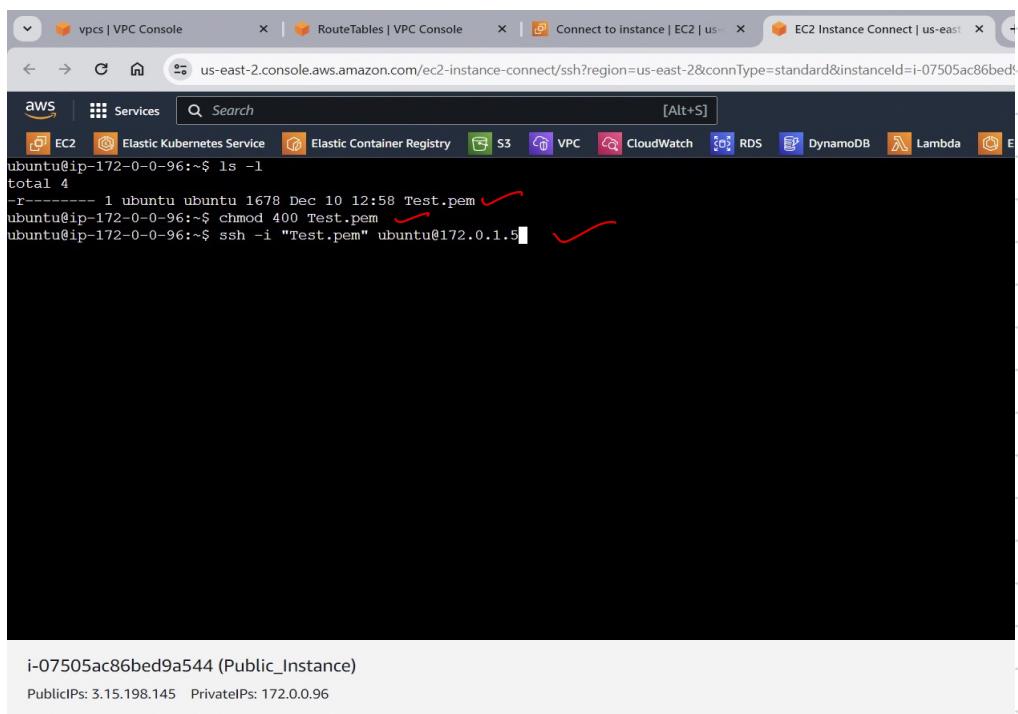
public IP to it. Try
to ping 8.8.8.8 from
there.

15. After creating instance
copy the .pem file
to connect from public
instance for that from
local shell run command

```
scp -i <filename to copy> username@ipadd:<path name where to copy>
```

```
hardy@DESKTOP-R359UB4 MINGW64 ~\Downloads
$ scp -i Test.pem "Test.pem" ubuntu@3.15.198.145:/home/ubuntu
The authenticity of host '3.15.198.145 (3.15.198.145)' can't be established.
ED25519 key fingerprint is SHA256:Ex5kLpqC42ruzgUuBlbGMKPRkFSvvQwcystl2u10MIK.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.15.198.145' (ED25519) to the list of known hosts.
Test.pem
```

16. After this Connect Public instance and on that look for the .pem file & change permissions



A screenshot of a terminal window within the AWS CloudWatch interface. The terminal shows the following command being run:

```
ubuntu@ip-172-0-0-96:~$ ls -l  
total 4  
-r----- 1 ubuntu ubuntu 1678 Dec 10 12:58 Test.pem  
ubuntu@ip-172-0-0-96:~$ chmod 400 Test.pem  
ubuntu@ip-172-0-0-96:~$ ssh -i "Test.pem" ubuntu@172.0.1.5
```

The command `chmod 400 Test.pem` is highlighted with a red box and a checkmark. The entire command line is also highlighted with a red box and a checkmark.

At the bottom of the terminal window, the instance ID and IP information are displayed:

i-07505ac86bed9a544 (Public_Instance)
PublicIPs: 3.15.198.145 PrivateIPs: 172.0.0.96

17. And then I shall
login to the private
subnet. And ping
8.8.8.8 from there

Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.
Enable ESM Apps to receive additional future security updates.
See <https://ubuntu.com/esm> or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Sun Dec 10 12:59:54 2023 from 172.0.0.96
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-0-1-5:~\$ 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=52 time=11.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=52 time=11.2 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=52 time=11.1 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=52 time=11.2 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss time 3005ms
rtt min/avg/max/mdev = 11.146/11.278/11.588/0.179 ms
ubuntu@ip-172-0-1-5:~\$

i-07505ac86bed9a544 (Public Instance)
Public IPs: 5.15.198.145 Private IPs: 172.0.0.96

This is IP of private instance

Activate Windows

Reach me on:

- 1 Github -> github.com/Hrmn97
- 2 Twitter -> twitter.com/Harman9765
- 3 LinkedIn -> linkedin.com/in/chetan-harman-56310424a
- 4 Website -> devhrmn.netlify.app

