

Assignment 1

Working for an organization, you are required to provide them with a safe and secure environment for the deployment of their resources. They might require different types of connectivity. Implement the following to fulfill the requirements of the company.

Tasks To Be Performed:

1. Create a VPC with 120.0.0.0/16 CIDR block.
2. Create 1 public subnet and 2 private subnets and make sure you connect a NAT gateway for internet connectivity to a private subnet

Step 1: Create VPC with the CIDR block 172.31.0.0/16

The screenshot shows the AWS VPC console. At the top, there's a search bar labeled "Find VPCs by attribute or tag". Below it, a table lists "Your VPCs (2) info" with columns: Name, VPC ID, State, Encryption controls, and Encryption control Two VPCs are listed: "vpc-05a20f627aab5c46" and "Assignment1-VPC". A "Create" button is visible at the top right. Below the table, another table shows the details of the VPCs, including their IPv4 CIDR blocks. The "Assignment1-VPC" has an IPv4 CIDR of "120.0.0.0/16".

Name	VPC ID	State	Encryption controls	IPv4 CIDR	IPv6 CIDR
vpc-05a20f627aab5c46	vpc-05a20f627aab5c46	Available	-	172.31.0.0/16	-
Assignment1-VPC	vpc-0305fce3b4a1528d8	Available	-	120.0.0.0/16	-

Step 2: Create 1 public subnet and 2 private subnets

The screenshot shows the AWS Subnet console. At the top, there's a search bar. Below it, a table lists subnets with columns: Subnet ID, State, VPC, Block Public..., and IPv4 CIDR. Three subnets are listed: "my-private-subnet1", "my-private-subnet2", and "my-public-subnet1". The "my-public-subnet1" has an IPv4 CIDR of "120.0.2.0/24".

Subnet ID	State	VPC	Block Public...	IPv4 CIDR
subnet-0bf17e081e6b85d47	Available	vpc-0305fce3b4a1528d8 Assi...	Off	120.0.2.0/24
subnet-0703129e1f849bdd4	Available	vpc-0305fce3b4a1528d8 Assi...	Off	120.0.3.0/24
subnet-0b35b65761733da8f	Available	vpc-0305fce3b4a1528d8 Assi...	Off	120.0.1.0/24

Step 3: Create the Internet Gateway and attach it to VPC

The screenshot shows the AWS CloudFormation console with the following details:

Details

- Internet gateway ID: igw-044d84032e87567bc
- State: Attached
- VPC ID: vpc-0305fce3b4a1528d8 | Assignment1-VPC
- Owner: 416946765337

Tags (1)

Key	Value
Name	Assignment-1-IGW

Actions

Step 4: Create the route table for public subnet

The screenshot shows the AWS CloudFormation console with the following details:

Details

- Route table ID: rtb-05d954063d14db8d9
- Main: No
- VPC: vpc-0305fce3b4a1528d8 | Assignment1-VPC
- Owner ID: 416946765337

Routes

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-044d84032e87567bc	Active	No	Create Route

Actions

The screenshot shows the AWS CloudFormation console with the following details:

Details

- Route table ID: rtb-05d954063d14db8d9
- Main: No
- VPC: vpc-0305fce3b4a1528d8 | Assignment1-VPC
- Owner ID: 416946765337

Subnet associations

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
my-public-subnet1	subnet-0b35b65761733da8f	120.0.1.0/24	-

Actions

Step 5: Create a NAT gateway

The screenshot shows the AWS VPC NAT gateways page. The top navigation bar includes the AWS logo, search bar, and account information (Account ID: 4169-4676-5337, Lokesh1234). On the left, a sidebar lists Subnets, Route tables, Internet gateways, and Egress-only internet gateways. The main content area displays a table titled 'NAT gateways (3) Info'. The table has columns: Name, NAT gateway ID, Connectivity type, State, State message, and Availability. One row is selected, showing 'NAT-Gateway' with NAT gateway ID 'nat-10e740859132a245e', Connectivity type 'Public', State 'Available', State message '—', and Availability 'Regional'.

Step 6: Create private route table and associate them with private subnets to it.

The screenshot shows the AWS Route Tables page. The top navigation bar includes the AWS logo, search bar, and account information (Account ID: 4169-4676-5337, Lokesh1234). The main content area shows a route table named 'rtb-00e9fc47cf78dd84 / Private-RT'. The 'Details' tab is selected, displaying information such as Route table ID, Main status (No), Owner ID, and Explicit subnet associations (2 subnets). Below this, tabs for Routes, Subnet associations, Edge associations, Route propagation, and Tags are visible. The 'Subnet associations' tab is active, showing 'Explicit subnet associations (2)'. It lists two subnets: 'my-private-subnet1' (subnet-0bf17e081e6b85d47, 120.0.2.0/24) and 'my-private-subnet2' (subnet-0703129e1f849bdd4, 120.0.3.0/24). An 'Edit subnet associations' button is present at the top right of this section.

Testing whether we can connect to the Internet through the public subnet

Create a EC2 instance and assign the VPC created use the public subnet assign auto assign public IP and launch the Ec2 instance

The screenshot shows the AWS EC2 Instances page. The top navigation bar includes the AWS logo, search bar, and account information (Account ID: 4169-4676-5337, Lokesh1234). The main content area shows a table titled 'Instances (1/3) Info'. A success message 'Successfully initiated termination (deletion) of i-0fd025b390b38b204' is displayed. The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Pub. One instance is listed: 'VPC-PROD' (i-0375855a81aaa92e5, Running, t3.micro, Initializing, View alarms, us-east-1a).

Connect from the outside using the ssh command to the Ec2 instance

And ping google.com

```
C:\Users\lokesh_ht>ssh -i "C:\Users\lokesh_ht\Downloads\aws_prac2.pem" ec2-user@3.92.18.147
The authenticity of host '3.92.18.147 (3.92.18.147)' can't be established.
ED25519 key fingerprint is SHA256:ow3u7N3T8kQHFhiCGC/mxdoNwUAUW0ZWcW0cTCzYLug.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.92.18.147' (ED25519) to the list of known hosts.

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

[ec2-user@ip-120-0-1-73 ~]$ ping google.com
PING google.com (192.178.155.101) 56(84) bytes of data.
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=1 ttl=102 time=2.13 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=2 ttl=102 time=2.16 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=3 ttl=102 time=2.15 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=4 ttl=102 time=2.17 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=5 ttl=102 time=2.24 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=6 ttl=102 time=2.14 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=7 ttl=102 time=2.17 ms
64 bytes from yuiadrs-in-f101.le100.net (192.178.155.101): icmp_seq=8 ttl=102 time=2.16 ms
^C
--- google.com ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7012ms
rtt min/avg/max/mdev = 2.133/2.165/2.241/0.031 ms
```

Assignment-2

Problem Statement:

Working for an organization, you are required to provide them a safe and secure environment for the deployment of their resources. They might require different types of connectivity. Implement the following to fulfill the requirements of the company.

Tasks To Be Performed:

1. Create 2 VPCs in the North Virginia region named MYVPC1 and MYVPC2

The screenshot shows the AWS VPC dashboard with the title "Your VPCs". It displays two VPCs: "MYVPC1" and "MYVPC2". Both VPCs are listed as "Available" with "Encryption controls" set to "-". The "Block Public..." setting is also listed as "Off". The "Actions" button is highlighted in orange.

Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public...
MYVPC1	vpc-0c25b0e151a956de5	Available	-	-	Off
MYVPC2	vpc-0cd261808beea97bd	Available	-	-	Off

2. Create one VPC in the Oregon region named VPCOregon1

The screenshot shows the AWS VPC dashboard with the title "Your VPCs". It displays one VPC named "VPCOregon1". This VPC is listed as "Available" with "Encryption controls" set to "-". The "Actions" button is highlighted in orange.

Name	VPC ID	State	Encryption c...	Encryption control ...
-	vpc-03c45d6088d776738	Available	-	-
VPCOregon1	vpc-084ca28b074c519fb	Available	-	-

3. Create a peering connection between MYVPC1 and MYVPC2

The screenshot shows the AWS VPC Peering connections page. On the left sidebar, under 'Virtual private cloud', there are several options: Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, and Managed prefix lists. The main content area is titled 'Peering connections (1) Info'. It displays a table with one row for the peering connection. The columns are: Name (Peer-MYVPC1-MYVPC2), Peering connection ID (pcx-0099b59cbcd7266e7), Status (Pending acceptance), Requester VPC (vpc-0c25b0e151a956de5 / MY...), and Acceptor VPC (vpc-0cd261808beea97bd / MY...). There are 'Actions' and 'Create peering connection' buttons at the top right.

This screenshot shows the same AWS VPC Peering connections page as the previous one, but the status of the peering connection has changed. The 'Status' column now shows 'Active' with a green checkmark. A green notification bar at the top states: 'Your VPC peering connection () 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.' There is also a 'Modify my route tables now' button. The rest of the interface is identical to the first screenshot.

4. Create a peering connection between MYVPC2 and VPCOregon1

This screenshot shows the AWS VPC Peering connections page with two entries in the table. The first entry is 'Peer-MYVPC1-MYVPC2' (Status: Active). The second entry is 'Peer-MYVPC2-Oregon' (Status: Pending acceptance). A green notification bar at the top states: 'A VPC peering connection pcv-07bb781e04cdc73dd / Peer-MYVPC2-Oregon has been requested. Remember to change your region to us-west-2 to accept the peering connection.' Below the table, the text 'pcx-0099b59cbcd7266e7 / Peer-MYVPC1-MYVPC2' is displayed. The rest of the interface is consistent with the previous screenshots.

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

Name	Peering connection ID	Status	Requester VPC	Acceptor VPC
-	pcx-07bb781e04cdc73dd	Pending acceptance	vpc-0cd261808beea97bd	vpc-084ca28b074c519fb

Peering connections (2) [Info](#)

Name	Peering connection ID	Status	Requester VPC	Acceptor VPC
Peer-MYVPC2-Oregon	pcx-07bb781e04cdc73dd	Active	vpc-0cd261808beea97bd / MY...	vpc-084ca28b074c519fb
Peer-MYVPC1-MYVPC2	pcx-0099b59cbcd7266e7	Active	vpc-0cd261808beea97bd / MY...	vpc-025b0e151a956de5 / MY...

Create route table for the traffic flow on VPC1 and VPC2

[Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Routes (2)

Destination	Target	Status	Propagated	Route Origin
10.0.0.0/16	local	Active	No	Create Route Table
10.1.0.0/16	pcx-0099b59cbcd7266e7	Active	No	Create Route

Create route table to Oregon VPC from VPC2

[Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Routes (2)

Destination	Target	Status	Propagated	Route Origin
10.1.0.0/16	local	Active	No	Create Route Table
10.2.0.0/16	pcx-07bb781e04cdc73dd	Active	No	Create Route

The screenshot shows the AWS VPC Route Tables page. At the top, there is a green success message: "Updated routes for rtb-0d4032cfe57dce1a3 successfully". Below this, the "Route table ID" is listed as "rtb-0d4032cfe57dce1a3", "Main" is set to "Yes", and "Owner ID" is "416946765337". The "VPC" section shows "vpc-084ca28b074c519fb | VPCOregon1". Below this, there are tabs for "Routes", "Subnet associations", "Edge associations", "Route propagation", and "Tags". The "Routes" tab is selected, displaying a table with two entries:

Destination	Target	Status	Propagated	Route Origin
10.1.0.0/16	pcx-07bb781e04cdc73dd	Active	No	Create Route
10.2.0.0/16	local	Active	No	Create Route Table

Assignment 2

Working for an organization, you are required to provide them with a safe and secure environment for the deployment of their resources. They might require different types of connectivity. Implement the following to fulfill the requirements of the company.

Tasks To Be Performed:

1. Create 2 VPCs in the North Virginia region named MYVPC1 and MYVPC2
2. Create one VPC in the Oregon region named VPCOregon1
3. Create a peering connection between MYVPC1 and MYVPC2
4. Create a peering connection between MYVPC2 and VPCOregon1

Step1 : Create two VPC's one in N virginia and one in oregon

The screenshot shows the AWS VPC dashboard for the 'United States (N. Virginia)' region. It displays two VPCs: 'MYVPC1' and 'MYVPC2'. Both VPCs are listed as 'Available' with their respective VPC IDs. The interface includes a search bar, filter options, and actions buttons for each VPC entry.

The screenshot shows the AWS VPC dashboard for the 'United States (Oregon)' region. It displays three VPCs: 'MYVPC1', 'MYVPC2', and 'VPCOregon1'. All three VPCs are listed as 'Available' with their respective VPC IDs. A success message at the top indicates the creation of 'VPCOregon1'. The interface includes a search bar, filter options, and actions buttons for each VPC entry.

Step 2: Create peering b/w VPC1 and VPC2

VPC → Peering Connections → Create

Name: Peering-VPC1-VPC2

Requester VPC: MYVPC1

Acceptor VPC: MYVPC2

Once the peering request is done from the N Virginia region accept the request for peering same on Oregon region.

Assignment 3

Problem Statement:

Working for an organization, you are required to provide them a safe and secure environment for the deployment of their resources. They might require different types of connectivity. Implement the following to fulfill the requirements of the company.

Tasks To Be Performed:

1. Create 2 EC2 instances in any public subnet of any VPC and name them Master and Client.
2. Using security groups, make sure that the Client instance can only be accessed (SSH) through the Master instance.

<input type="checkbox"/> Client	i-09a04539a99fcf76d	Running	t3.micro	3/3 checks passed View alarms +	us-east-1a	-
<input checked="" type="checkbox"/> Master	i-06eed8725b934fd6a	Running	t3.micro	3/3 checks passed View alarms +	us-east-1a	ec2-

-06eed8725b934fd6a (Master)

Details	Status and alarms	Monitoring	Security	Networking	Storage	Tags			
Instance summary Info <table border="1"> <tr> <td>Instance ID i-06eed8725b934fd6a</td> <td>Public IPv4 address 3.236.129.242 open address</td> <td>Private IPv4 addresses 172.31.7.184</td> </tr> </table>							Instance ID i-06eed8725b934fd6a	Public IPv4 address 3.236.129.242 open address	Private IPv4 addresses 172.31.7.184
Instance ID i-06eed8725b934fd6a	Public IPv4 address 3.236.129.242 open address	Private IPv4 addresses 172.31.7.184							

<input checked="" type="checkbox"/> Client	i-09a04539a99fcf76d	Running	t3.micro	3/3 checks passed View alarms +	us-east-1a	-
<input type="checkbox"/> Master	i-06eed8725b934fd6a	Running	t3.micro	3/3 checks passed View alarms +	us-east-1a	ec2-

i-09a04539a99fcf76d (Client)

Details	Status and alarms	Monitoring	Security	Networking	Storage	Tags			
Instance summary Info <table border="1"> <tr> <td>Instance ID i-09a04539a99fcf76d</td> <td>Public IPv4 address -</td> <td>Private IPv4 addresses 172.31.4.176</td> </tr> </table>							Instance ID i-09a04539a99fcf76d	Public IPv4 address -	Private IPv4 addresses 172.31.4.176
Instance ID i-09a04539a99fcf76d	Public IPv4 address -	Private IPv4 addresses 172.31.4.176							

Connect to the Master Ec2 using the public Ip

```
ec2-user@ip-172-31-7-184:~ % + ^

C:\Users\lokesht> ssh -i "C:\Users\lokesht\Downloads\aws_prac2.pem" ec2-user@3.236.129.242
               _#
               ~\_ #####_      Amazon Linux 2023
               ~~ \_#####\
               ~~ \###|
               ~~ \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
               ~~ \~' `-->
               ~~~ .-
               ~~~ /`-/
               _/m/'|_
Last login: Sun Jan 25 09:27:19 2026 from 49.37.171.188
[ec2-user@ip-172-31-7-184 ~]$ |
```

Copy the pem key to the linux m/c of master instance

```
PS C:\Users\lokesh_ht> scp -i "C:\Users\lokesh_ht\Downloads\aws_prac2.pem" "C:\Users\lokesh_ht\Downloads\aws_prac2.pem"
ec2-user@3.236.129.242:~/aws_prac2.pem
                                                100% 1834      7.5KB/s   00:00
PS C:\Users\lokesh_ht>
```

Change the permission for pem key

```
Last login: Sun Jan 25 09:45:50 2026 from 49.37.171.188
[ec2-user@ip-172-31-7-184 ~]$ chmod 400 aws_prac2.pem
[ec2-user@ip-172-31-7-184 ~]$
```

Connect to the Client Ec2 instance using the private key

```
[ec2-user@ip-172-31-7-184 ~]$ ssh -i aws_prac2.pem ec2-user@172.31.4.176
'_
#_
~\_ #####_          Amazon Linux 2023
~~ \_#####\_
~~   \###|
~~     \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '-->
~~_
~~_. _/
~~/_ /_
~/m/'_
Last login: Sun Jan 25 09:44:02 2026 from 172.31.7.184
[ec2-user@ip-172-31-4-176 ~]$ |
```

Assignment 4

Problem Statement:

Working for an organization, you are required to provide them a safe and secure environment for the deployment of their resources. They might require different types of connectivity. Implement the following to fulfill the requirements of the company.

Tasks To Be Performed:

1. Create a VPC endpoint for a S3 bucket of your choice for secure access to the files.

Create a Gateway VPC Endpoint for S3

Endpoints (1) Info					
Create endpoint					
Actions ▼					
Name	VPC endpoint ID	Endpoint type	Status	Service name	
My-endpoint01	vpce-0a6f34feff2123f9d	Gateway	Available	com.amazonaws.us-east-1.s3	

Choose the route tables associated with the subnets where your EC2 instances run.

The screenshot shows the AWS VPC Endpoint configuration page for endpoint ID vpce-0a6f34feff2123f9d. The 'Route tables' tab is selected, displaying a table with one entry:

Name	Route Table ID	Main	Associated Id
-	rtb-067de3095dde75e9b	Yes	subnet-0a08d7d636dcc1f06 (public-s...)

Attach a policy (IAM-like JSON) to control access. Example: allow full S3 access:

```
{  
  "Statement": [  
    {  
      "Action": "*",  
      "Effect": "Allow",  
      "Resource": "*",  
      "Principal": "*"  
    }  
  ]  
}
```

In the S3 bucket policy, restrict access to only requests coming via the VPC endpoint.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Principal": "*",  
      "Action": "s3:PutObject",  
      "Resource": "arn:aws:s3:::mybucket/*",  
      "Condition": {"aws:SourceVpc": "vpce-0a6f34feff2123f9d"}  
    }  
  ]  
}
```

```
"Action": "s3:*",
"Effect": "Allow",
"Resource": [
    "arn:aws:s3:::my-secure-bucket",
    "arn:aws:s3:::my-secure-bucket/*"
],
"Condition": {
    "StringEquals": {
        "aws:SourceVpce": "vpce-1234567890abcdef0"
    }
}
}
}
]
}
```

Test Connectivity

```
An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
[ec2-user@ip-172-31-7-184 ~]$ [ec2-user@ip-172-31-7-184 ~]$ aws s3 ls s3://my-secure-bucket --region us-east-1
An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied^C
```

Assignment 5

Problem Statement:

You work for XYZ Corporation and based on the expansion requirements of your corporation you have been asked to create and set up a distinct Amazon VPC for the production and development team. You are expected to perform the following tasks for the respective VPCs.

Production Network:

1. Design and build a 4-tier architecture.
2. Create 5 subnets out of which 4 should be private named app1, app2, dbcache and db and one should be public, named web.
3. Launch instances in all subnets and name them as per the subnet that they have been launched in.
4. Allow dbcache instance and app1 subnet to send internet requests.
5. Manage security groups and NACLs.

Development Network:

1. Design and build 2-tier architecture with two subnets named web and db and launch instances in both subnets and name them as per the subnet names.
2. Make sure only the web subnet can send internet requests.
3. Create peering connection between production network and development network.
4. Setup connection between db subnets of both production network and development network respectively.