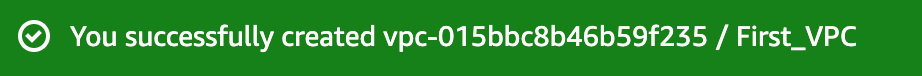
**Lab Steps**

Task 1: Sign in to AWS Management Console

1. Click on the  button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
   * Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. otherwise, you cannot proceed with the lab.
   * Now copy your **User Name** and **Password** in the Lab Console to the **IAM Username and Password** in AWS Console and click on the **Sign in** button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as **US East (N. Virginia) us-east-1.**

Task 2: Create the first VPC

1. Make sure you are in the N.VirginiaRegion.
2. Navigate to **VPC** by clicking on the **Services** menu in the top, then click on **VPC** in the **Networking & Content Delivery** section.
3. Navigate to **Your VPCs** on the left panel and click on the .
   * Select **VPC only**
   * Name tag : Enter ***First\_VPC***
   * IPv4 CIDR block : Enter ***10.0.0.0/24***
4. Leave everything else as default and click on the .
5. You have successfully created the VPC. Note the VPC ID for later use.



1. Now select the **First\_VPC** from the list and click on the  and select **Edit VPC Settings**
2. Check the **Enable DNS resolution**and **Enable DNS hostnames**checkbox under**DNS settings** and then click on the **Save** button.

Graphical user interface, application

Description automatically generated

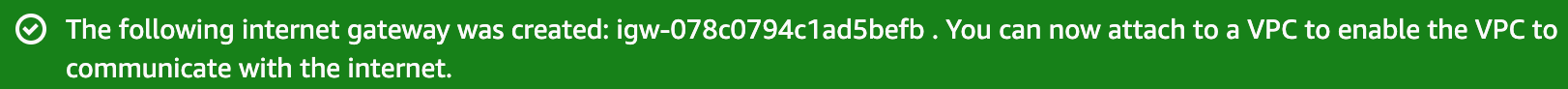
Task 3: Create a Public subnet in First VPC

1. Navigate to **Subnet** from the left side menu and click on button.
   * VPC ID : Select the **First\_VPC** VPC from the list.
   * Subnet name : Enter ***Public\_subnet\_first\_VPC***
   * Availability Zone : Leave as No Preference
   * IPv4 CIDR block : Enter ***10.0.0.0/25***
2. Now click on the .



Task 4: Create and attach an Internet Gateway

1. Navigate to the **Internet gateways** from the left side menu and click on the .
   * Name tag : Enter ***IGW***
2. Click on the 



1. Now click on the  and select .
   * Available VPCs :  select **First\_VPC** from the list.
2. Now click on the .



Task 5: Create a Public Route Table and associate it with the subnet

1. Navigate to **Route tables** on the left side panel and click on the A picture containing logo

   Description automatically generated
   * Name : Enter ***PublicRT***
   * VPC\* : Select the **First\_VPC** from the list.
2. Now click on the A picture containing logo

   Description automatically generated.
3. Switch to the A picture containing logo

   Description automatically generatedtab in below.
4. Click on the Text

   Description automatically generated.
5. Now select the subnet with name **Public\_subnet\_first\_VPC** and click on the A picture containing rectangle

   Description automatically generated.

Task 6: Add public Route in the Route table

1. Navigate to **Route tables** on the left side panel and select the **PublicRT** from the list.
2. Switch to the  tab in below and click on the A picture containing shape

   Description automatically generated.
3. Now click on the A picture containing shape

   Description automatically generated
   * Destination : Enter ***0.0.0.0/0***
   * Target : select **Internet Gateway** and then select the Internet Gateway id present.

Graphical user interface, application

Description automatically generated

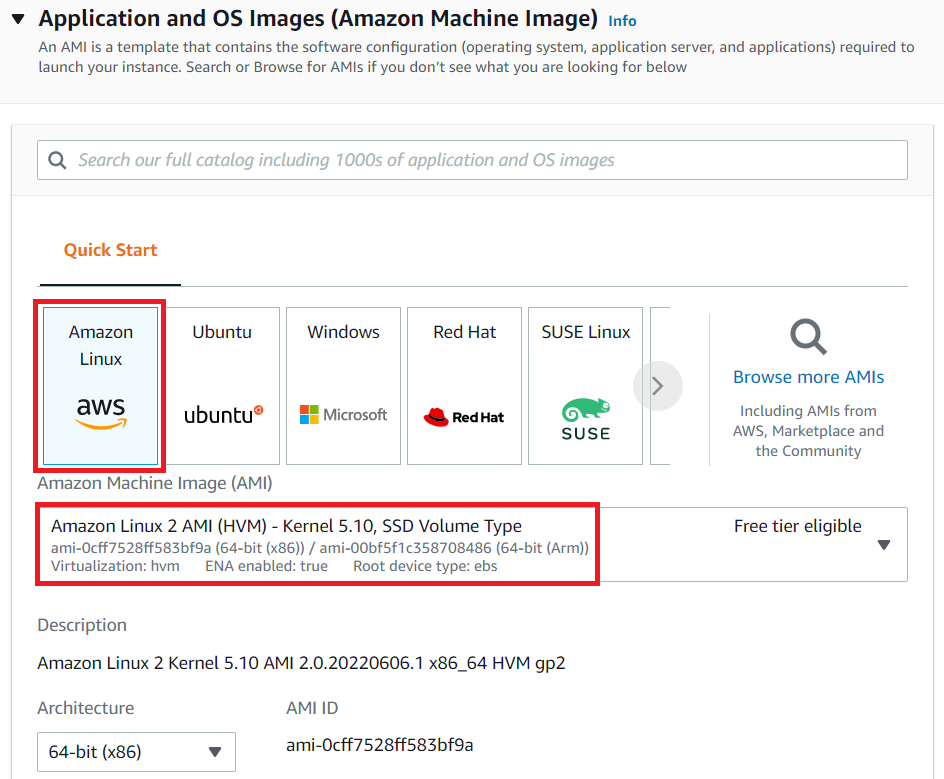
1. Click on the .

Task 7: Launch an EC2 instance in the First VPC

1. Make sure you are in the **N. Virginia(us-east-1)** Region.
2. Navigate to **EC2** by clicking on the **Services** menu in the top left, then click on **EC2** in the **Compute** section.
3. Navigate to **Instances** from the left side menu and click on **Launch Instances** button.
4. Under the **Name and tags** section :
   * Name : ***First\_VPCs\_EC2***
5. Under the **Application and OS Images (Amazon Machine Image)** section :

* Select **Quick Start** tab and **Amazon Linux** under it
* Amazon Machine Image (AMI) : select ***Amazon Linux 2 AMI***

**Note: if there are two AMI's present for Amazon Linux 2 AMI, choose any of them.**

****

1. Under the **Instance Type** section **:**

* Instance Type : Select **t2.micro**

**Graphical user interface, application

Description automatically generated**

1. Under the **Key Pair (login)** section **:**

* Click on **Create new key pair** hyperlink
* Key pair name: ***ec2\_ssh\_key***
* Key pair type: **RSA**
* Private key file format: **.pem**
* Click on **Create key pair** and select the created key pair

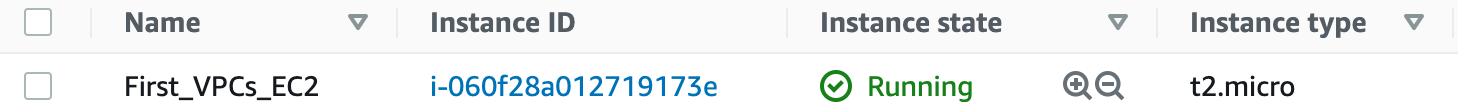
1. Under the **Network Settings** section **:**

* Click on **Edit** button
* VPC : Select **First\_VPC**
* Subnet : leave as default
* Auto-assign public IP: select ***Enable***
* Firewall (security groups) : Select **Create a new security group**
* Security group name : Enter ***Public\_EC2\_SG***
* Description : Enter ***Security group for public EC2***
* To add **SSH:**
  + Choose Type: **SSH**
  + Source: **Anywhere** (From ALL IP addresses accessible).
* For **HTTP**, click on **Add security group rule**,
  + Choose Type: **HTTP**
  + Source: **Anywhere**  (From ALL IP addresses accessible).
* For **HTTPS**, click on **Add security group rule**,
  + Choose Type: **HTTPS**
  + Source: **Anywhere** (From ALL IP addresses accessible).

1. Under the **Advanced details** section **:**
   * Under the **User data:** copy and paste the following script to create an HTML page served by an Apache httpd web server.

|  |
| --- |
| #!/bin/bash  sudo su  yum update -y  yum install httpd -y  systemctl start httpd  systemctl enable httpd  echo "<html><h1> Welcome to Whizlabs Public Server</h1><html>" > /var/www/html/index.html |

1. Keep everything else as default and click on the **Launch instance** button.
2. **Launch Status:** Your instance is now launching, Navigate to **Instances** page from the left menu and wait until the status of the EC2 Instance changes to **running**.



1. Note down the sample IPv4 Public IP Address of the EC2 instance. A sample is shown in the screenshot below.

Graphical user interface, application

Description automatically generated

1. If you paste the IPv4 Public IP in your browser and hit [enter]. You will be able to the below webpage.

Graphical user interface, text

Description automatically generated

Task 8: Create a Second VPC

1. Navigate to **VPC** by clicking on the **Services** menu at the top, then click on **VPC** in the **Networking & Content Delivery** section.
2. Navigate to **Your VPCs** on the left panel and click on the .
   * Select **VPC only**
   * Name tag : Enter ***Second\_VPC***
   * IPv4 CIDR block : Enter ***20.0.0.0/24***
3. Leave everything else as default and click on the .
4. You have successfully created the VPC. Note the VPC ID for later use.



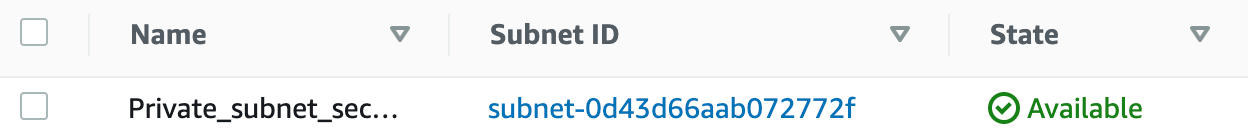
1. Now select the **Second\_VPC** from the list and click on the  and select **Edit VPC settings**
2. Check the **Enable DNS resolution** and **Enable DNS hostnames**checkbox under DNS settings and then click on the **Save** button

Task 9: Create a Private subnet in Second VPC

1. Navigate to **Subnets** from the left side menu and click on button.

* VPC ID : Select the **Second\_VPC** VPC from the list.
* Subnet name : Enter ***Private\_subnet\_second\_VPC***
* Availability Zone : Leave as No Preference
* IPv4 CIDR block : Enter ***20.0.0.0/25***

2. Now click on the .



Task 10: Launch an EC2 instance in Second VPC

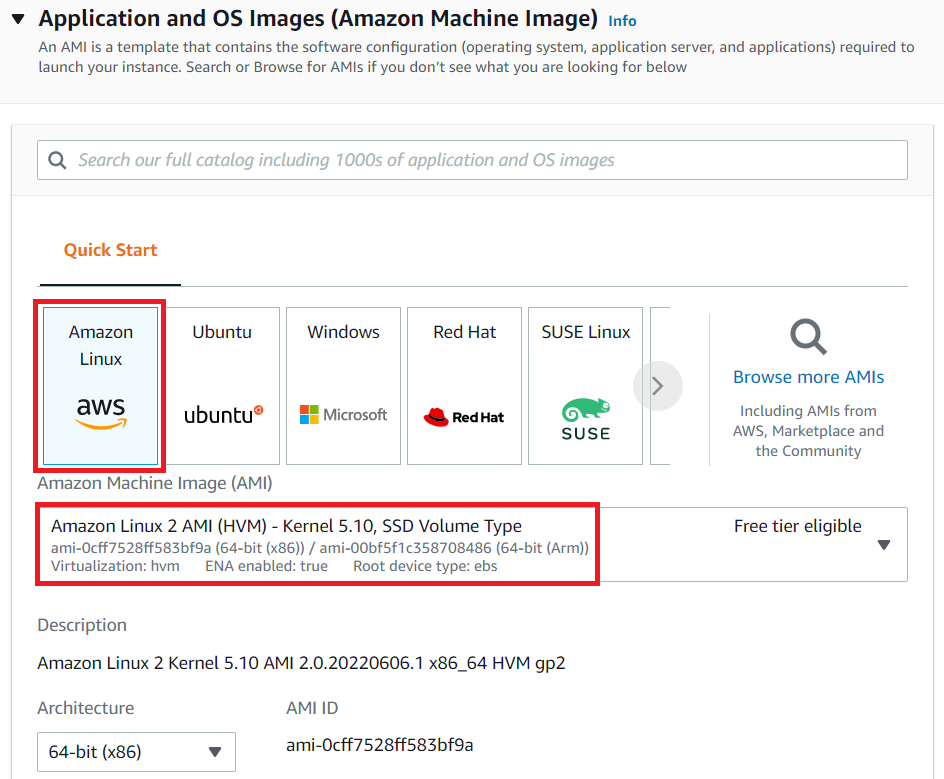
1. Now again click on **Launch Instances** button.
2. Under the **Name and tags** section :

* Name : ***Second\_VPCs\_EC2***

1. Under the **Application and OS Images (Amazon Machine Image)** section :

* Select **Quick Start** tab and **Amazon Linux** under it
* Amazon Machine Image (AMI) : select ***Amazon Linux 2 AMI***

**Note: if there are two AMI's present for Amazon Linux 2 AMI, choose any of them.**

****

1. Under the **Instance Type** section **:**
   * Instance Type : Select **t2.micro**

**Graphical user interface, application

Description automatically generated**

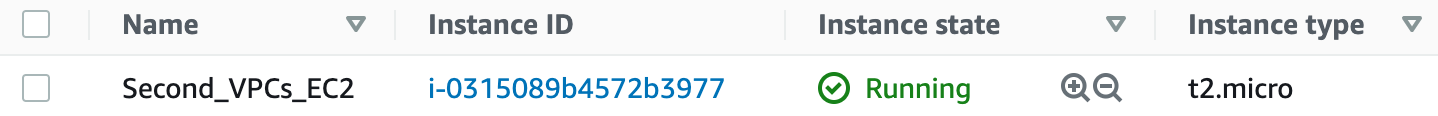
1. Under the **Key Pair (login)** section **:**

* Select **ec2\_ssh\_key** from the list.

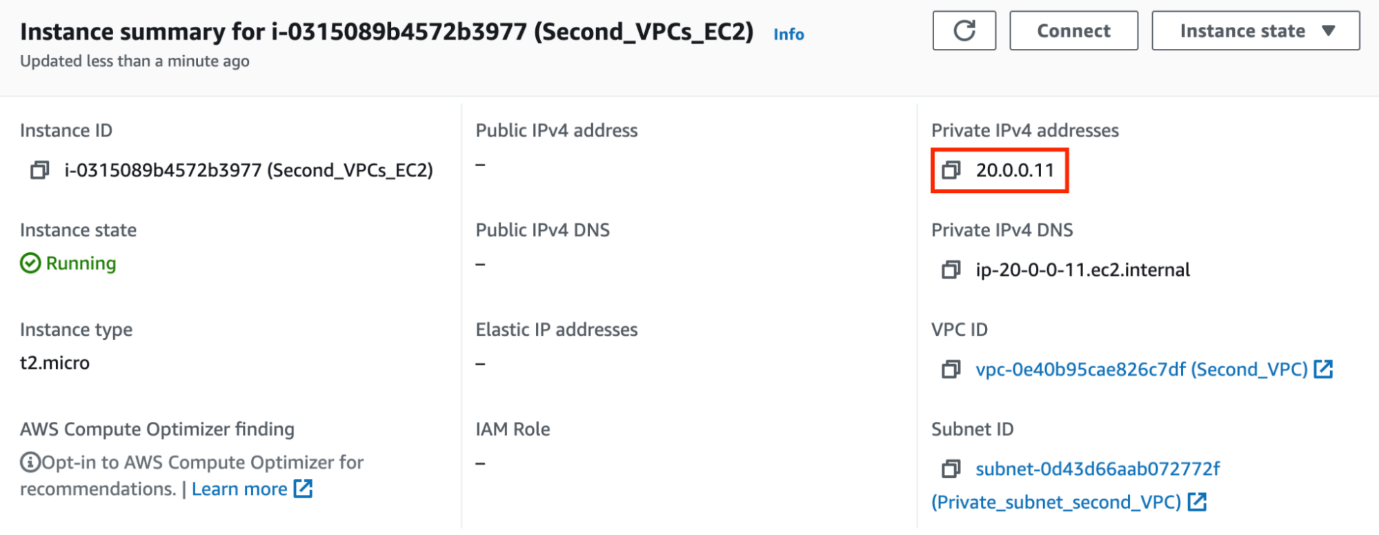
1. Under the **Network Settings** section **:**

* Click on Edit button
* VPC : Select **Second\_VPC**
* Subnet : leave as default
* Auto-assign public IP: select ***Disable***
* Firewall (security groups) : Select **Create a new security group**
* Security group name : Enter ***Private\_EC2\_SG***
* Description : Enter ***Security group for private EC2***
* To add **SSH:**
  + Choose Type: **SSH**
  + Source: **Anywhere** (From ALL IP addresses accessible).

1. Keep everything else as default and then click on the **Launch Instance** button.
2. Your instances are now launching. Navigate to the EC2 instance page and wait until the status changes to the **Running**. It will usually take 1-2 minutes.



Since this EC2 is created in a private subnet, the machine will only have Private IP so, note down the sample IPv4 Private IP Address of the EC2 instance. A sample is shown in the screenshot below.



Task 11: Create a Transit gateway

1. Navigate to **VPC** by clicking on the **Services** menu at the top, then click on **VPC** in the **Networking & Content Delivery** section.
2. Click on the   present under Logo

   Description automatically generated with medium confidencesection on the left sidebar.
3. Click on the  to create a Transit gateway.
   * Name tag: Enter ***DemoTG***
   * Description: Enter ***TG for peering two VPCs***
4. Keep all the options as default and click on 
5. Currently, the status of the Transit gateway is in a **pending** state. It takes up to 5 minutes for it to become **available**.

Graphical user interface, text, application

Description automatically generated

Task 12: Create two Transit gateway attachment for the VPCs created

1. Navigate to the  **Transit gateways attachments** present under Logo

   Description automatically generated with medium confidencesection on the left side bar.
2. Click on the 
   * Name tag : **First\_VPC\_TGA**
   * Transit Gateway ID: Select transit gateway present with Name tag **DemoTG.**
   * Attachment type: Select **VPC**
   * DNS support: **Checked (default)**
   * IPv6 support: **Unchecked**
   * VPC ID: Select VPC with the Name **First\_VPC**
   * Subnet IDs: **Default**
3. Click on the 
4. Creation will be in-progress for the Transit gateway attachment.
5. It takes upto 5 minutes for it to come in **available** state.
6. To create the Transit gateway attachment for the second VPC, Click on the 
   * Name tag: **Second\_VPCs\_TGA**
   * Transit Gateway ID: Select transit gateway present with Name tag **DemoTG.**
   * Attachment type: Select **VPC**
   * DNS support: **Checked (default)**
   * IPv6 support: **Unchecked**
   * VPC ID: Select VPC with the Name **Second\_VPC**
   * Subnet IDs: **Default**
7. Creation will be in-progress for the Transit gateway attachment.
8. Once created both the Transit gateway attachment will be present.

Graphical user interface, text, application

Description automatically generated

Task 13: Add the routes in the First VPC’s route table

1. Navigate to **Route tables** on the left side panel.
2. There will be 4 route tables present, and to avoid confusion put the entry in the correct route table. Let's filter the route table present using the **VPC filter**.
3. To get the filter using the VPC option, simply click on the search bar and it will show the different options for the filter. Select **VPC** from that list and choose the **ID of First VPC**.  
   Graphical user interface, application

   Description automatically generated
4. For the First VPC you created, you also have created a Route table with the name **PublicRT** which has a subnet association. Here the **PublicRT** route table is also called a custom or non-default route table.
5. Click on the **Route table ID** to see the routes present.  
   Graphical user interface, application, Teams

   Description automatically generated
6. Click on the  tab below and click on the A picture containing shape

   Description automatically generated.
7. For this Route table, there are two entries present, First the local entry i.e. CIDR block of the First VPC, the second entry is about the route to the internet with Destination as 0.0.0.0/0 having target as the Internet gateway.
8. Let’s add the third route which has destination as **20.0.0.0/24** i.e. CIDR range of second VPC and Target as Transit gateway.
9. To add the third route, Click on the A picture containing shape

   Description automatically generated.
   * Destination: Enter ***20.0.0.0/24***
   * Target:  Enter ***Transit Gateway***  
     Graphical user interface, application

     Description automatically generated
10. Click on the .
11. Make sure Routes have Destination as 20.0.0.0/24  
    Graphical user interface, text, application, email, Teams

    Description automatically generated
12. **Note: Sometimes it does not show the destination as 20.0.0.0/24, in that case, please add the following from Step 6 of this task.**

Task 14: Add the routes in the Second VPC’s route table

1. Navigate to **Route tables** on the left side panel.
2. Filter the VPC’s by using the ID of the second VPC.
3. In the second VPC, we have not created any extra route table, so there will be only one route table present which was created during the creation of VPC itself and it’s called a default route table or main route table.
4. Click on the **Route table ID** to see the routes present.  
   Graphical user interface, application

   Description automatically generated
5. Click on the  tab in below and click on the A picture containing shape

   Description automatically generated.
6. There will be only one entry to local, as we have not created an internet gateway because this is a private route table.
7. Let’s add the entry to ***10.0.0.0/24*** i.e. CIDR of the first VPC as a destination and Transit gateway as a target.
8. To get the **Transit gateway ID**, click on the search button and select **Transit Gateway** from the list of options present.  
   Graphical user interface, application

   Description automatically generated
9. Click on the .
10. Make sure Routes have Destination as 10.0.0.0/24  
    Graphical user interface, text, application

    Description automatically generated
11. **Note: Sometimes it does not show the destination as 10.0.0.0/24, in that case, please add the following from Step 5 of this task.**

Task 15: Test the connectivity between two VPCs

1. You have copied the IPv4 Public IP of the EC2 instance created in the First VPC.
2. Please follow the steps in [SSH into EC2 Instance](https://www.whizlabs.com/labs/support-document/ssh-into-ec-instance).
3. Once you have successfully SSH in to EC2, run the following commands :

* Switch to root user :

sudo su

* Update server repository :

yum update -y

1. Now we need to copy the .pem key of the EC2 instance created.

* Create a file :

nano ec2\_ssh\_key.pem



* Open the .pem key of EC2 **ec2\_ssh\_key** in your local editor and paste it in the terminal file.  
  Text

  Description automatically generated
* Press **[ctrl] + x / [control] + x**
* Press **y** key in your keyboard.
* File Name : No changes, press **[Enter]** key in your keyboard

1. Change the .pem key permission

* chmod 400 ec2\_ssh\_key.pem

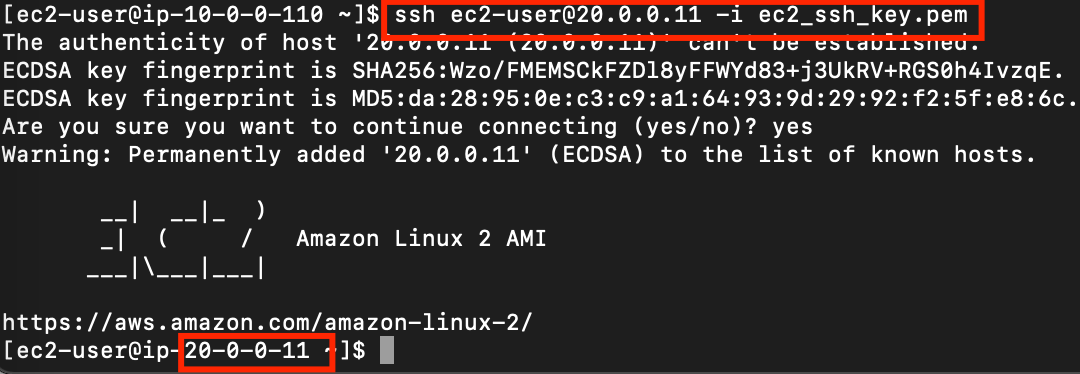
1. SSH into the Private EC2 **ec2\_ssh\_key**

* ssh ec2-user@<IPv4 private Ip> -i ec2\_ssh\_key.pem
* Copy the Private IP of **Second\_VPCs\_EC2**
* Example : **ssh ec2-user@20.0.0.11 -i ec2\_ssh\_key.pem**

1. If the connection prompts a message to confirm connect enter **yes**



1. As you can see the IP address is changed to private ec2 private IP 30.0.1.154



1. Now you have connected two VPCs using the Transit gateway.

Task 16: Validation Test

1. Once the lab steps are completed, please click on the button on the left side panel.
2. This will validate the resources in the AWS account and displays whether you have completed this lab successfully or not.
3. Sample output :

Graphical user interface, text, application, email

Description automatically generated

Task 17: Delete AWS Resources

Deleting EC2 Instances

1. Make sure you are in the **US East (N. Virginia)** Region.
2. Navigate to **EC2**by clicking on the **Services** menu in the top left, then click on **EC2** in the **Compute** section.
3. Now select both the EC2 instances that you have created, click on **Instance State** and click on the **Terminate instance** option.

Graphical user interface, application

Description automatically generated

1. Click on the **Terminate** button and your EC2 will start terminating.

**Completion and Conclusion**

1. You have successfully created a VPC with a public subnet & internet gateway and Launched an EC2 instance.
2. You have successfully created a VPC with a private subnet and Launched an EC2 instance.
3. You have successfully created the Transit gateway.
4. You have successfully created the Transit gateway attachments for both the VPC’s.
5. You have successfully tested the connectivity of VPC after peering using the Transit gateway.

**End Lab**