Introduction to Amazon Elastic Compute Cloud (EC2)

Level: **Fundamental**

Amazon EC2 Amazon Web Services

0h 27m 28s left

15 MIN

End Lab

Open Console

Validation

Your last attempt on **23-Sep-2025** View all

Overview Steps Validation

Cloud Architect

Compute

Lab Steps

Task 1: Sign in to AWS Management Console

Lab Credentials

User Name

Whiz_User_58170.99117657

Password

287b47f2-2052-450f-b

Access Key

Help

1. Click on the **Open Console** 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: Provision Default VPC

1. Navigate to **VPC** by clicking on the **Services** menu in the top, then click on **VPC** or Open the Amazon VPC console via https://console.aws.amazon.com/vpc/.
2. Delete the **default VPC** by following the **below steps:**
   - In the navigation pane, choose **Your VPCs.**
   - Select the **VPC** with value as **yes** in **default VPC column.**

   - Go to **actions button** and click on **delete VPC button.**

AKIA4Z6RE4OMEIWQJMMP

Secret Key

bJhNp9+kxYhp3INjExwlrkh

Lab Resources

No Lab Resources Found

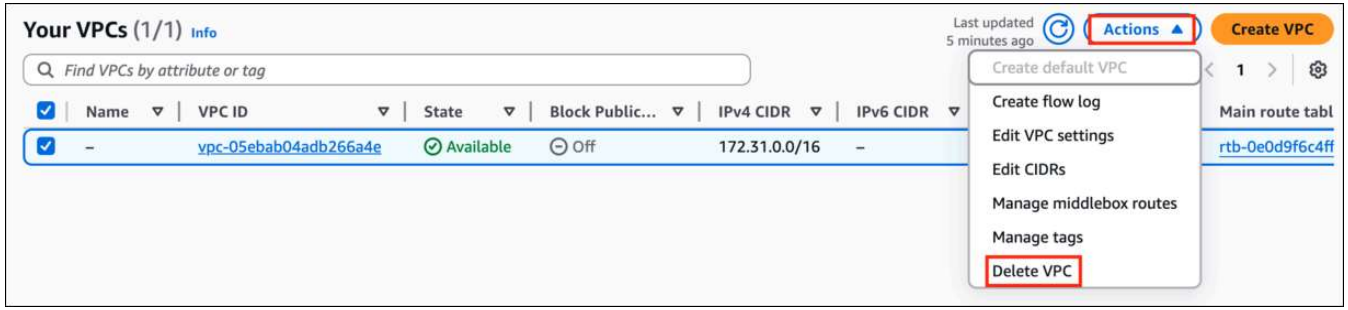
Support Documents

1. FAQs and Troubleshooting
2. SSH into EC2 Instance

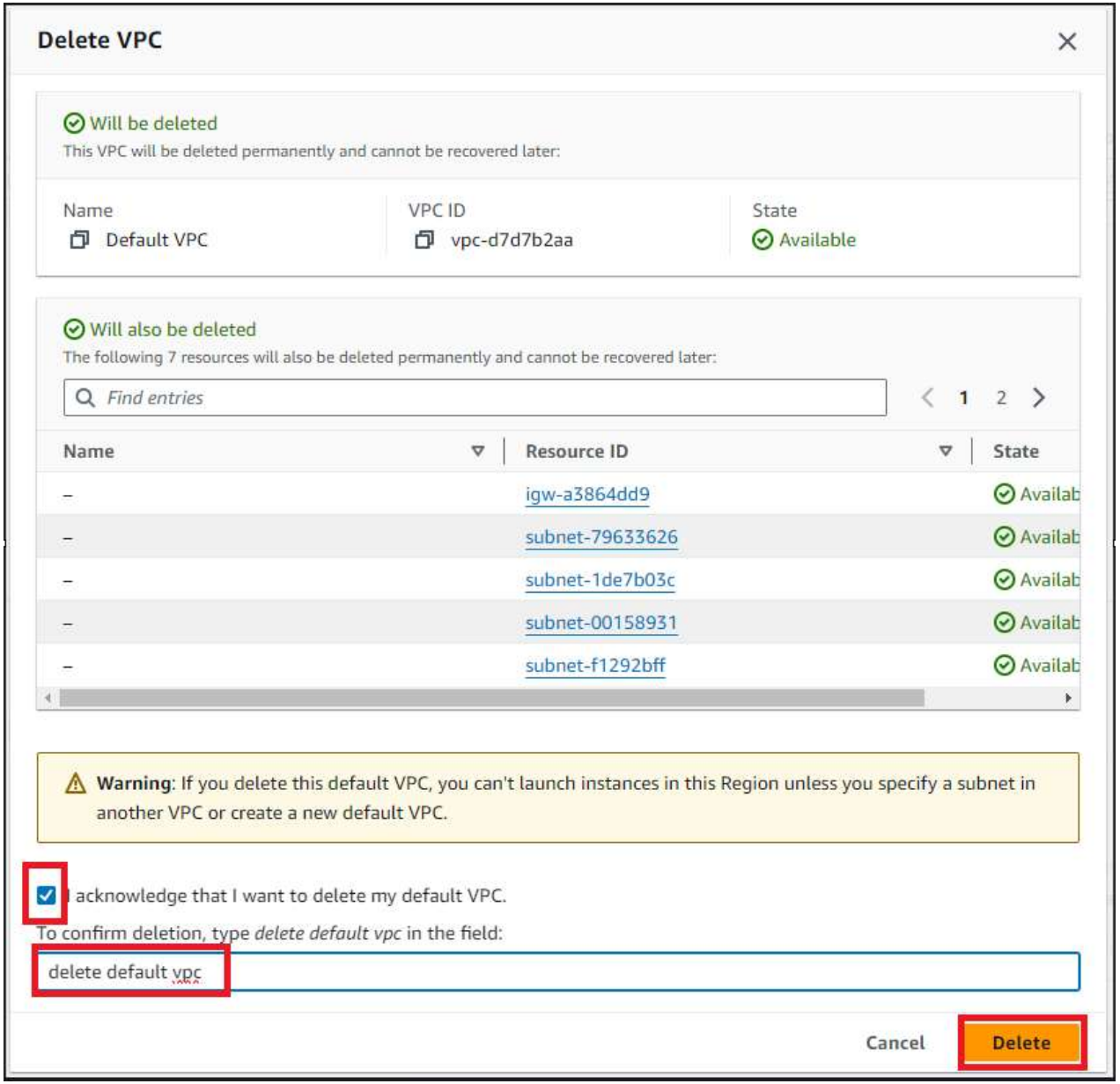
Need help?

- How to use Hands on Lab
- Troubleshooting Lab
- FAQs

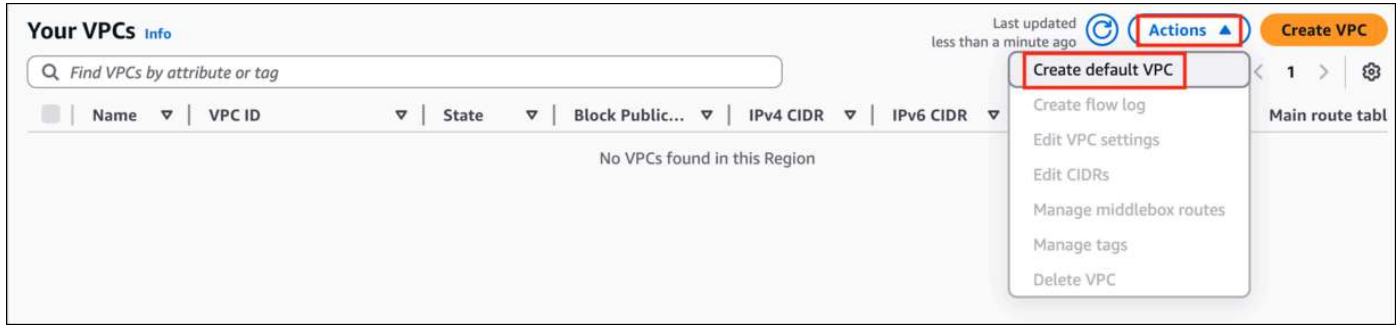
Submit Feedback | Share



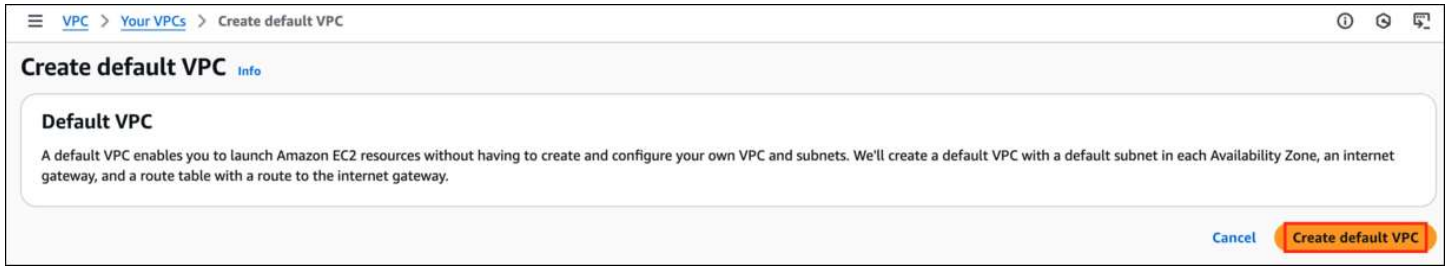
- Check **I acknowledge that I want to delete my default VPC** option.
- Type default delete VPC and click on **Delete button.**

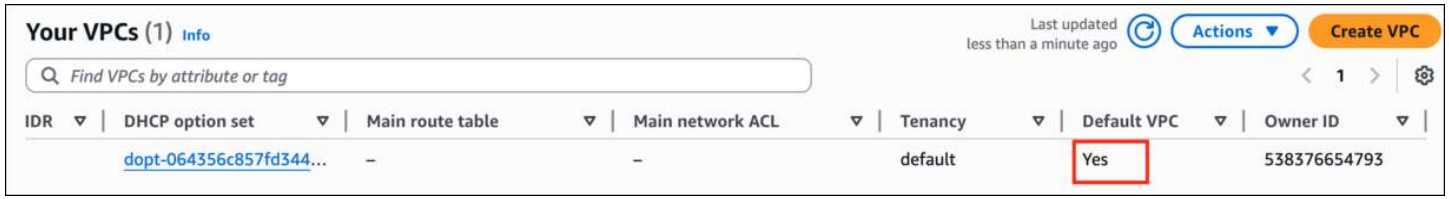


3. Now to provision **Default VPC** again, Refresh your console go to **actions** and click **Create default VPC**



4. Click **Create default VPC button** and your default VPC will get created

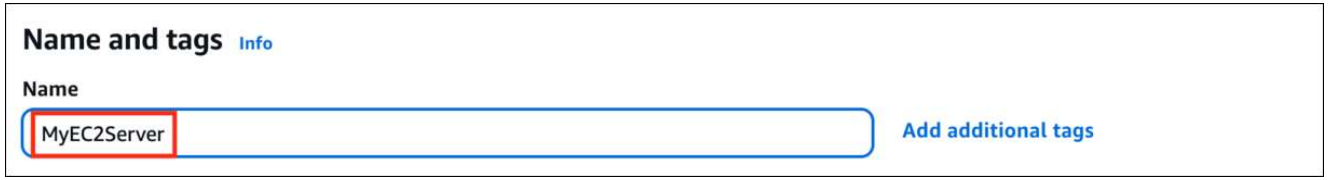




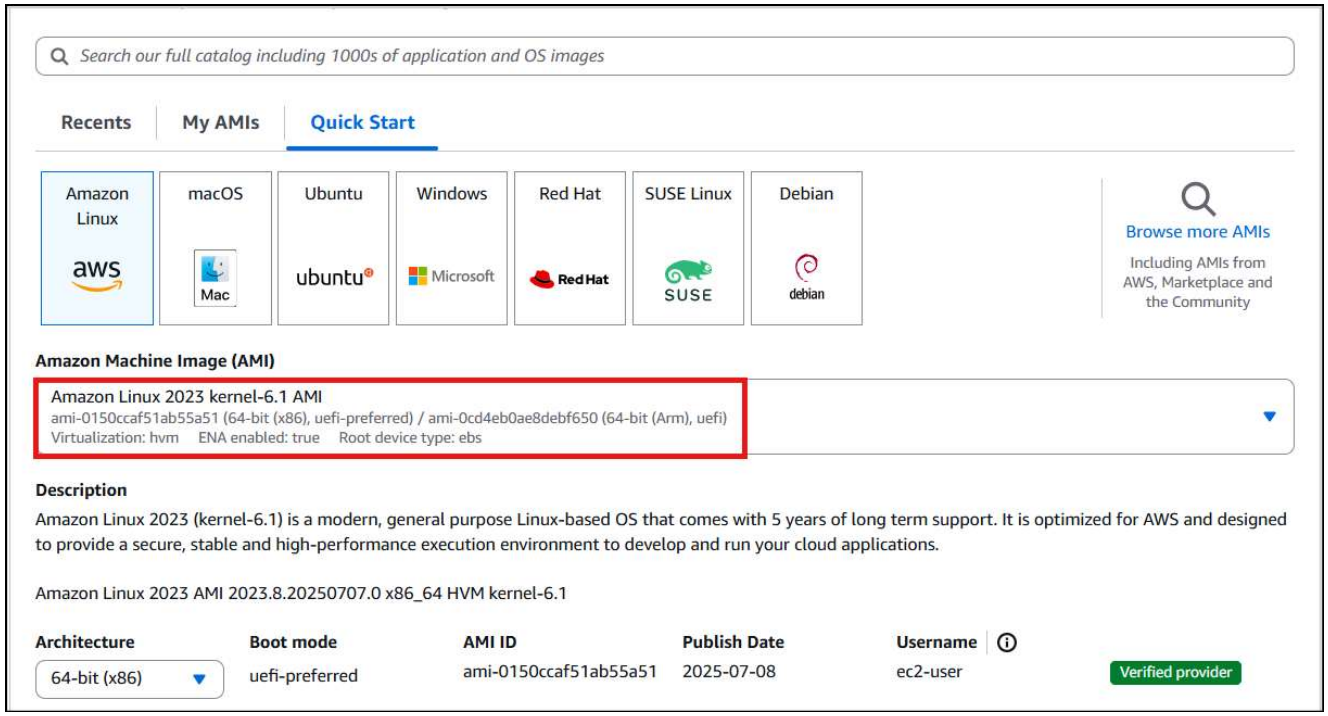
Task 3 : Launch an EC2 Instance with desired specifications

1. Ensure you are in the **US East (N. Virginia) us-east-1** Region to begin launching an EC2 instance in the Amazon cloud.
2. Navigate to **EC2** by clicking on the **Services** menu in the top, then click on **EC2** in the **Compute** section.
3. Click on the **Instances** option on the left panel, and then click on the **Launch Instances** button.

4. Name : Enter **MyEC2Server**



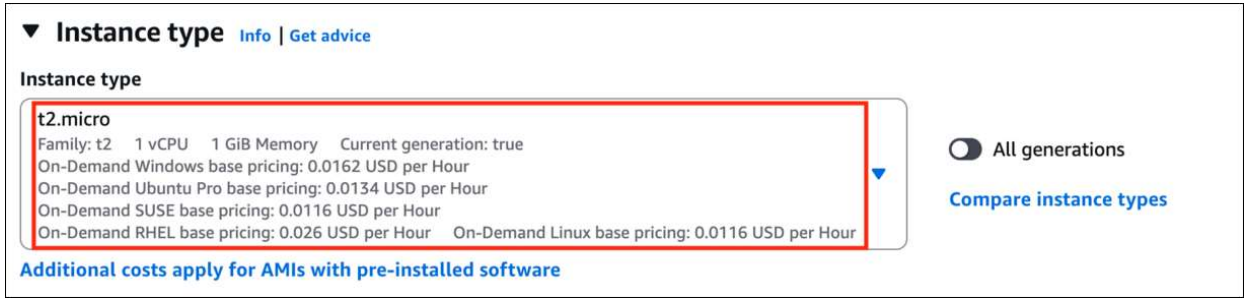
5. Select **Amazon Linux 2023 AMI** from the dropdown.



Note: if there are two AMI's present for **Amazon Linux 2023 kernel-6.1 AMI**.

6. An instance type in AWS refers to a virtual server configuration that determines the computing resources, such as CPU, memory, and storage, available to an instance. It is the basic building block for creating an EC2 instance in the AWS cloud.

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



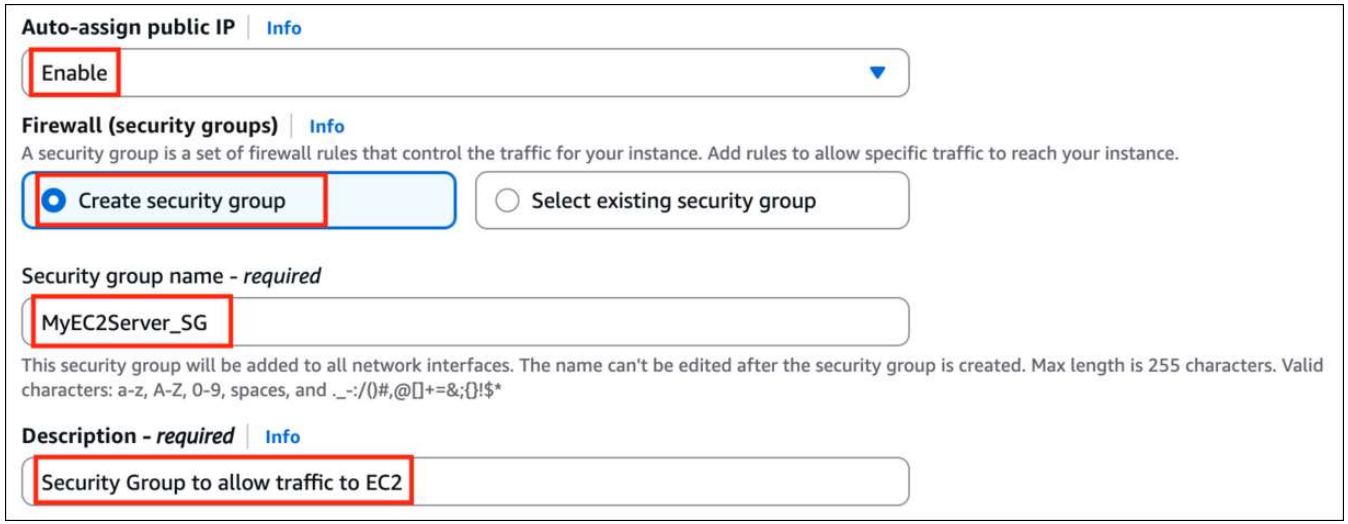
- t2.micro is an instance type in AWS that comes with 1 vCPU, and 1GB memory and is suitable for low-traffic web servers, small development environments, and other lightweight applications.

7. AWS key pair is a secure pair of keys used for login and access to EC2 instances. It includes a public key placed on the instance and a private key kept on the user's local computer, used for authentication to prevent unauthorized access.

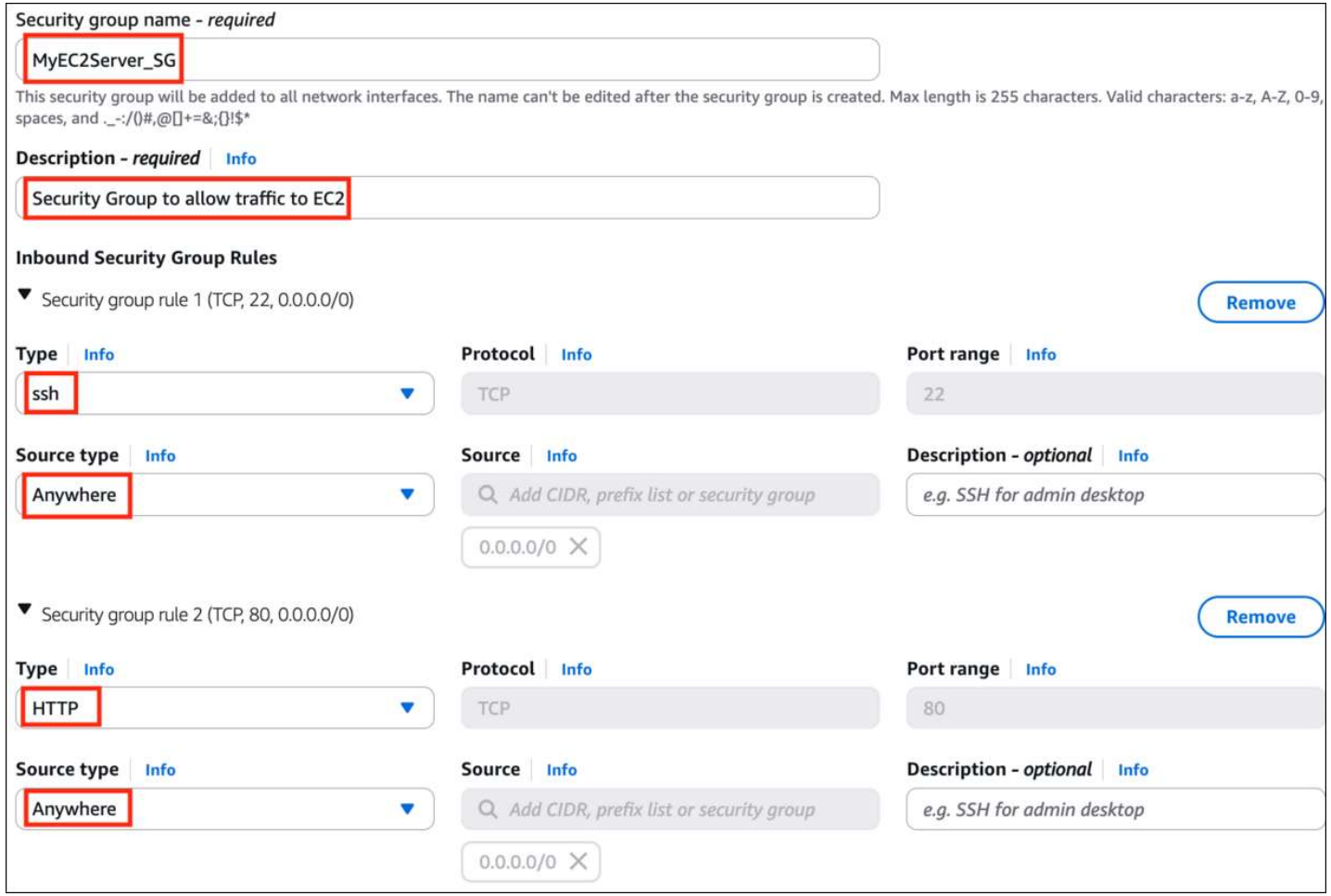
- For **Key pair(login)**: Select **Create a new key pair** Button
  - Key pair name: **WhizKey**
  - Key pair type: **RSA**
  - Private key file format: **.pem**

8. In **Network Settings,** Click on **Edit** Button:

- Auto-assign public IP: **Enable**
- Select **Create security group**
- Security group name: Enter **MyEC2Server_SG**
- Description: Enter **Security Group to allow traffic to EC2.**



9. We will now add the security group rules. SSH will already be present there.
   - For HTTP, Select **Add security group rule** Button
     - Choose Type: Select **HTTP**
     - Source: Select **Anywhere**

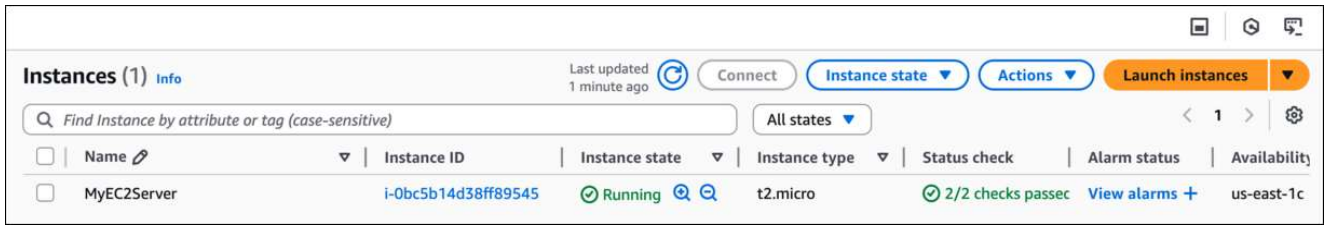


10. A security group is a virtual firewall that controls the inbound and outbound traffic for instances in a particular network in a cloud computing environment. Here we have selected SSH and HTTP rules that will allow incoming SSH and HTTP traffic to instances that are associated with the security group.

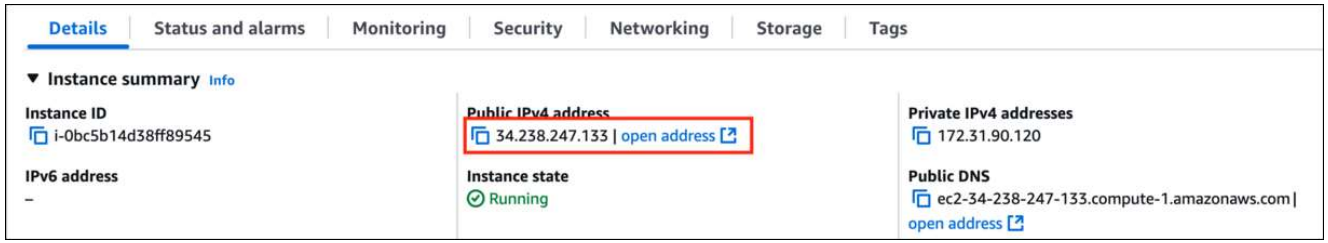
11. Proceed with launching the instance while leaving all other settings as default. Simply click on the **Launch Instance** without modifying any other configuration.

12. To view the instance that you have created, choose the **View all Instances** option.

13. **Launch Status:** Once you have initiated the instance launch process, Go to the Instances page from the left menu and wait for your EC2 instance to become "**Running**" while ensuring the health check status is **2/2 checks passed** for optimal performance.

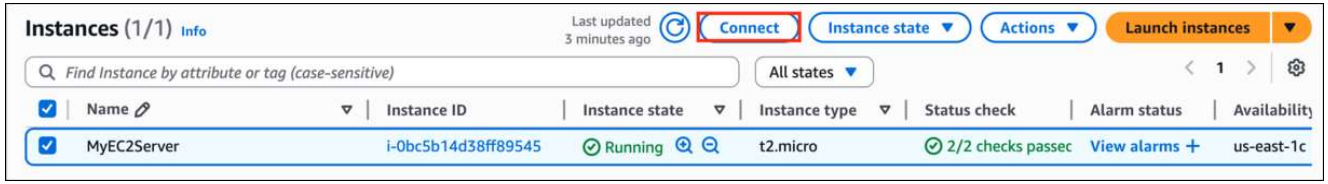


14. **Select** the instance that you have created and **copy** the public IPv4 address within the details section and **paste** it into the editor for later use. An example of this process is depicted in the screenshot provided.



Task 4 : SSH into EC2 Instance using the key pair

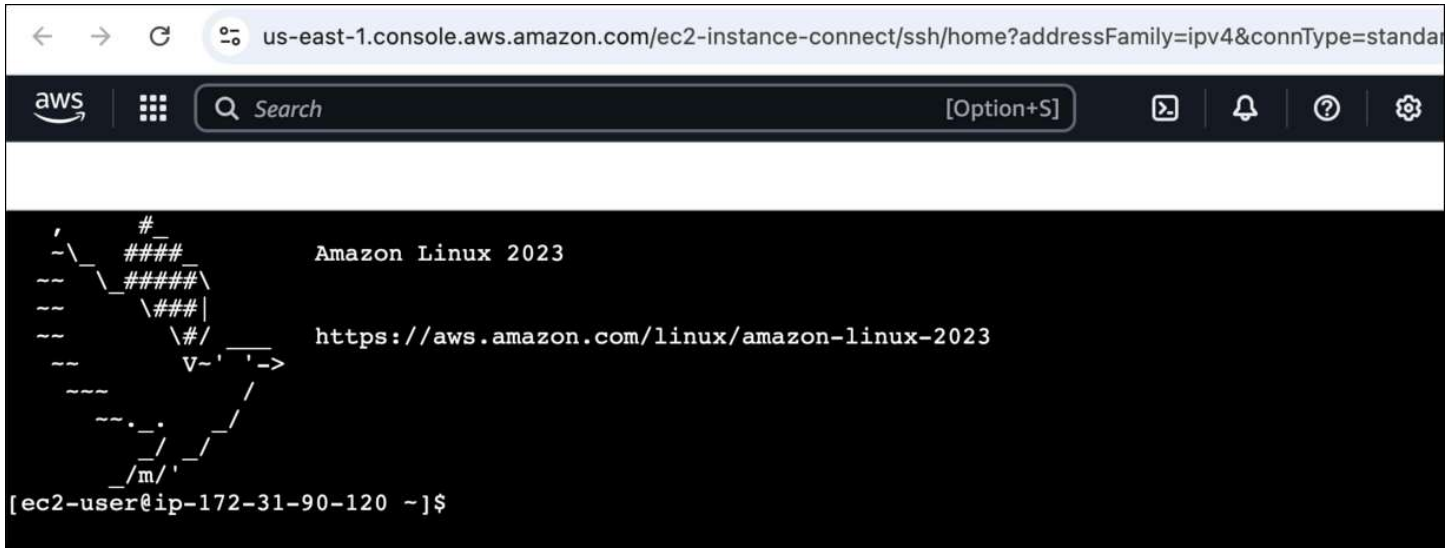
1. Select your EC2 instance **(MyEC2Server)** and click on the **Connect** button.



2. Select **EC2 Instance Connect** option and click on **Connect** button. Keep everything else as default.

3. A new tab will open in the browser where you can execute the Linux Commands.

4. Please follow the steps in SSH into EC2 Instance for more options to SSH.



Task 5: Install an Apache Server on the instance

In this task, our goal is to configure an Amazon EC2 instance to run an Apache Web Server and verify its functionality by accessing the web server via a web browser using the instance's public IPv4 address.

1. Switch to root user:

```
sudo su
```

2. Now run the updates using the following command:

```
dnf update -y
```

3. Once completed, lets install and run an apache server

4. Install the Apache web server:

```
dnf install httpd -y
```

5. Start the web server:

```
systemctl start httpd
```

6. Now Enable httpd:

```
systemctl enable httpd
```

7. Check the webserver status

```
systemctl status httpd
```

8. You can see Active status is running.

9. You can test that your web server is properly installed and started by entering the **public IPv4 address** of your **EC2 instance** in the address bar of a web browser. If your web server is running, then you see the Apache test page. If you don't see the Apache test page, then verify whether you followed the above steps properly and check your inbound rules for the security group that you created.

Task 6 : Create a web page and publish it on the instance

In this task, you will add content to the index.html file using the "echo" command and restart the webserver. Then, you can view the content by entering the public IPv4 address followed by "/index.html" in a web browser, ensuring that the URL protocol is HTTP.

1. To add the contents into index.html file using echo, copy and paste the below command to shell.

```
echo "<html>Hi Whizlabs, I am a public page</html>" > /var/www/html/index.html
```

2. Restart the webserver by using the following command:

```
systemctl restart httpd
```

3. Now enter the file name, ***/index.html*** after the **public IPv4 Address** which you got when you created the ec2 instance in the browser, and you can see your HTML content.

- Make sure **URL Protocol** is **http** not https.
- Syntax: **http://<Your_Public_IPv4_Address>/index.html**

- Sample URL: **http://52.87.50.168/*index.html***
- **Note:** If the index.html page is not loading, try removing **s** from the link, it should be HTTP.

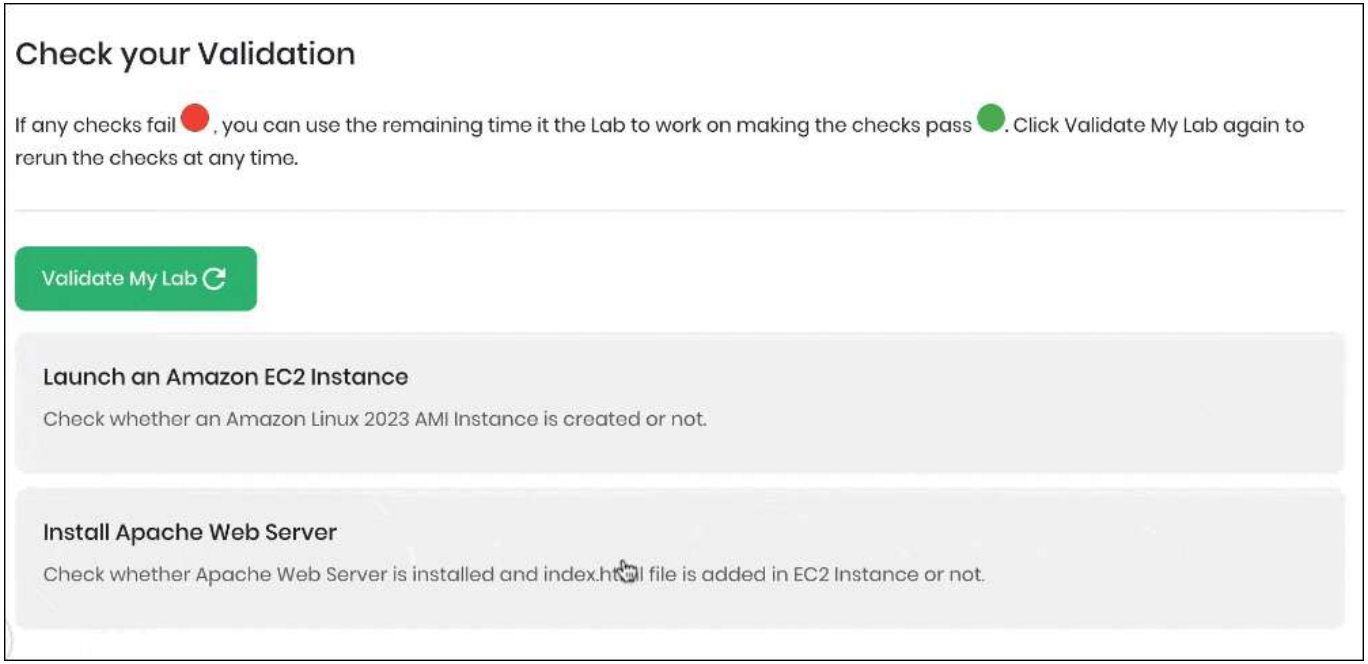
4. If you can see the above text in the browser, then you have successfully completed the lab.

Do You Know?

The **Amazon EC2 P4d instance** is the most powerful EC2 instance type available, with eight NVIDIA A100 Tensor Core GPUs, 1.1 TB of NVMe storage, and 400 Gbps network bandwidth, making it ideal for large-scale machine learning and HPC workloads.

Task 7 : Validation of the Lab

1. Once the lab steps are completed, please click on the **Validation** 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 :



Completion and Conclusion

1. You have successfully created and launched an Amazon EC2 instance
2. You have successfully logged into the EC2 instance using SSH
3. You have successfully installed an Apache server on the instance
4. You have successfully created a webpage and published it.
5. You have successfully validated the lab.

End Lab

1. Sign out of AWS Account.
2. You have successfully completed the lab.
3. Once you have completed the steps, click on **End Lab** from your Whizlabs lab console and wait till the process gets completed.

Java Certifications

HashiCorp Certified Terraform Associate Certification

Snowflake SnowPro Advanced Architect Certification

Docker Certified Associate

Need help? You can or +91 6364678444