**MULTI-TIER, HIGHLY AVAILABLE, FAULT-TOLERANT**

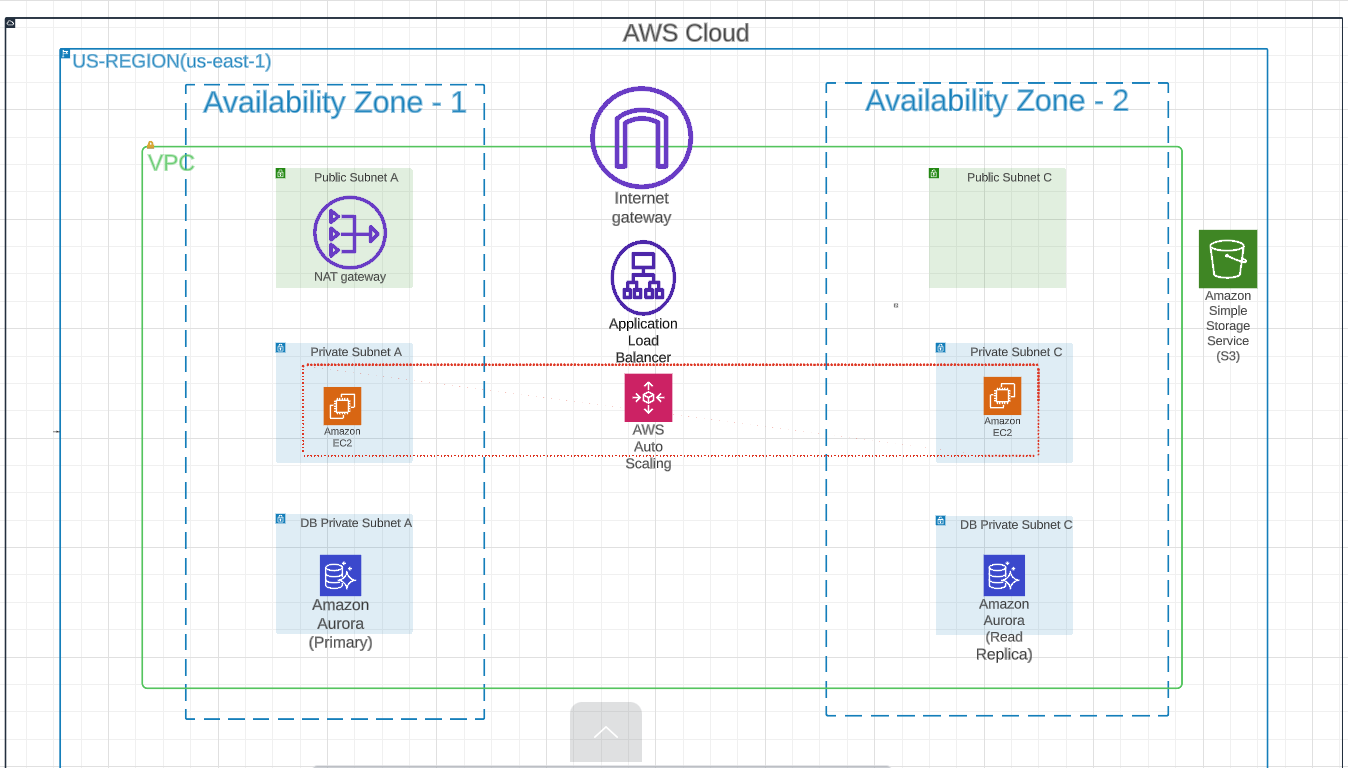
**WEB APPLICATION**

A.) Start with AWS ACCOUNT -

1. Creating an IAM USER -

**Create an AWS account**, create an **IAM user** that has access to AWS account. Log in to the AWS account, you can create a IAM user using IAM console. Create an IAM user with Administrator role as below the step. If you already have an IAM user with administrator role, skip the next task.

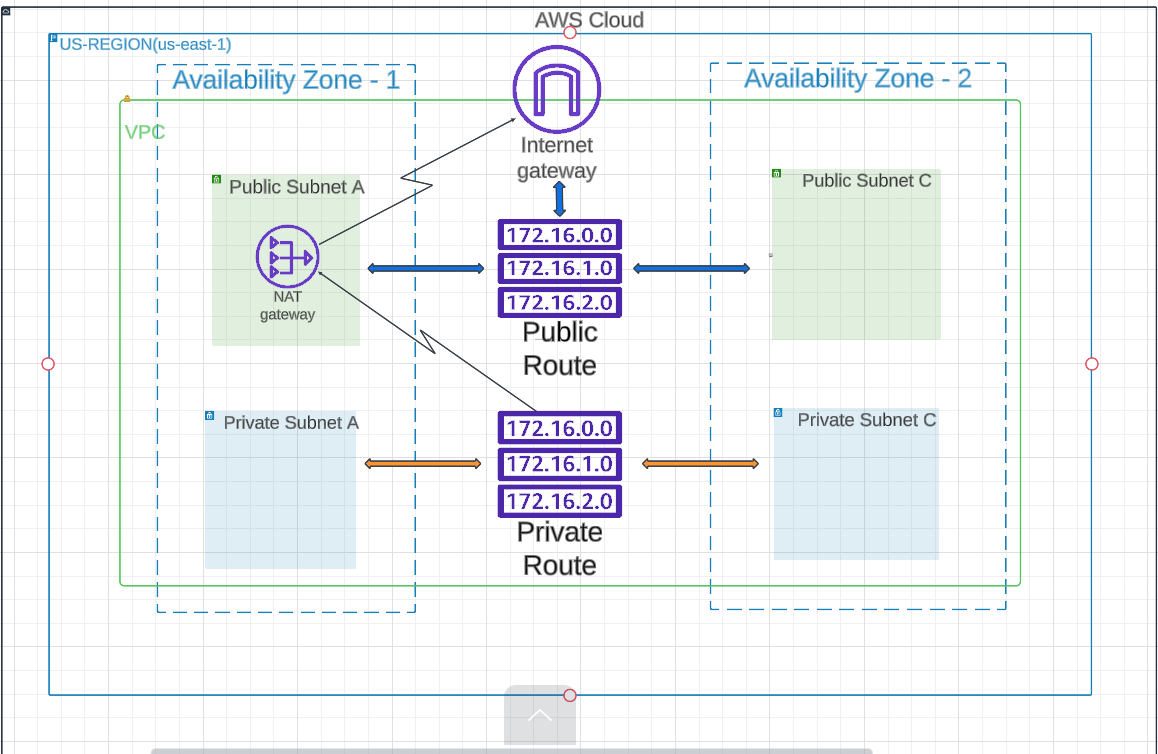
1. From the [Log in Page](https://console.aws.amazon.com/) , log in with AWS root account email address and password then go to [IAM Console](https://console.aws.amazon.com/iam/home#/home) .
2. From the left sidebar of the IAM Console, click **Users** and then click **Add user**
3. Enter **User name** as Administrator.
4. Tick **Provide user access to the AWS Management Console** check box.
5. Choose **I want to create an IAM user**
6. Choose **Custom password** and enter a password of your choice. You may need to remember this password so be sure to write it down or copy it for later.
7. Uncheck the **Users must create a new password** box. This will simplify the learning for this lab.
8. Click **Next.**
9. Choose Attach policies directly, tick AdministratorAccess option and then click Next.
10. Review AdministratorAccess, AWS managed policy, is added to the Administrator user and then click Create.
11. After creating the user, copy the login URL. The format of the URL is as below. https://<your\_aws\_account\_id>.signin.aws.amazon.com/console.
12. At this stage, you may also download the .csv file to save your password and any additional login information
13. Log out from the root user and then log in to the Administrator user you just created using the URL you copied above.



B.) Network – Amazon VPC

In this lab, we'll create not only one Public Subnet and one Private Subnet in two Availability Zone(AZ-a, AZ-c) but also one NAT Gateway placed on the Public Subnet through VPC Wizard. After configuring these resources, you will set up a routing table to define the network traffic flow. Through these tasks, you can complete a basic networking configuration to create a highly available and scalable web service environment.

NETWORK ARCHITECTURE



1.) Create VPC - **Amazon Virtual Private Cloud (Amazon VPC)** allows you to start AWS resources with a user-defined virtual network. This virtual network, along with the benefits of using AWS's scalable infrastructure, is very similar to the existing network operating in the customer's own data center.

1. After logging in to the AWS console, select VPC from the service menu.
2. Select VPC Dashboard and click Launch VPC Wizard to create your own VPC.
3. To create a space to provision AWS resources used in this lab, we will create a VPC and Subnets. Select VPC, subnets, etc in Resource to create tab and change name tag to VPC-Lab. Leave the default setting for IPv4 CIDR block.
4. To design high availability architecture, we create 2 subnet space and select 2a and 2c for Customize AZs. And set the CIDR value of the public subnet that can communicate directly with the Internet as shown in the screen below. Set the CIDR value of the private subnet as shown in the screen.
5. You can use a NAT gateway so that instances in your private subnets can connect to services outside your VPC, but external services cannot initiate direct connections to these instances. In this lab, we will create a NAT gateway in only one Availability Zone to save cost. Also, for DNS options, enable both DNS hostnames and DNS resolution. After confirming the setting value, click the Create VPC button.
6. As the VPC is created, you can see the process of creating network-related resources as shown in the screen below. For NAT Gateway, provisioning may take longer compared to other resources.
7. You can check the information of the created VPC. Check related information such as CIDR value, route table, network ACL, etc. Check that the values you just set are correct.

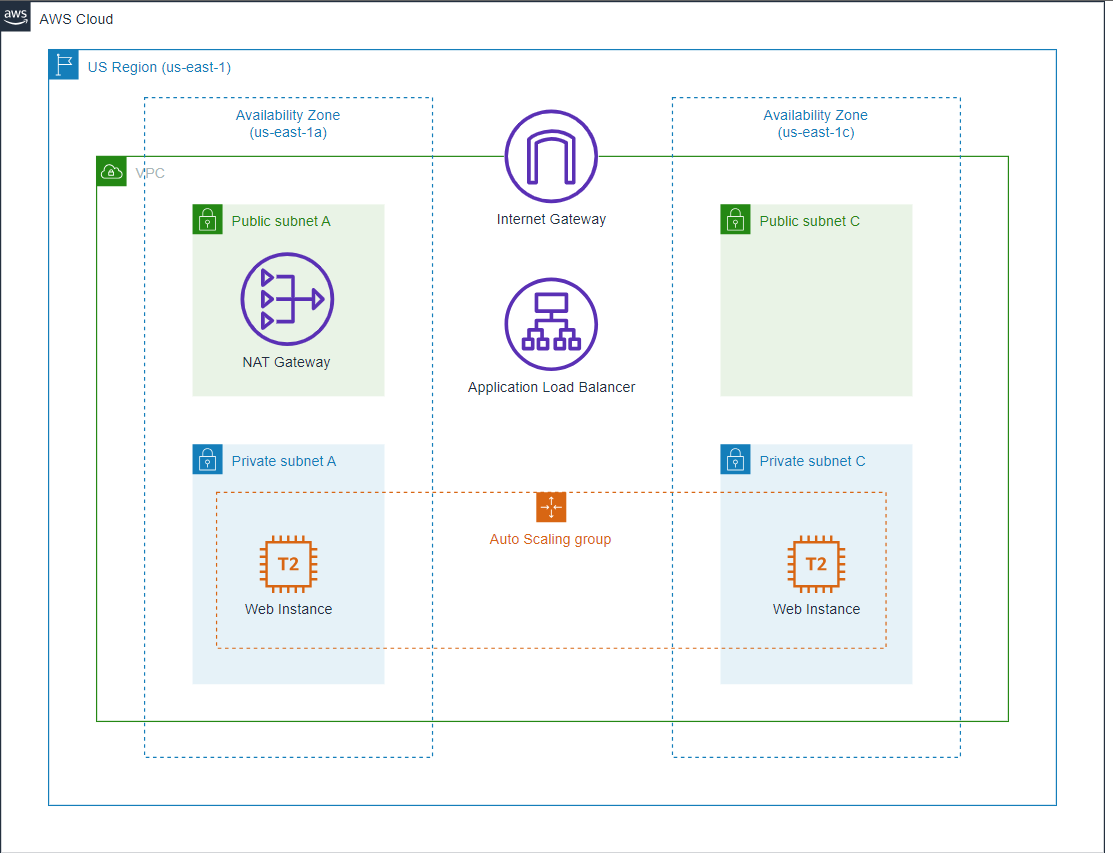
2.) Create VPC Endpoint - In this section, you create an endpoint for S3 to learn a VPC endpoint.

1. In VPC Dashboard, select Endpoints. Click Create endpoint button.
2. Type s3 endpoint for name and select AWS services in Service category tab. In the search bar below, type s3 and select the list at the top.
3. For S3 VPC endpoints, there are gateway types and interface types. For this lab, select the gateway type. And for the deployment location, select the VPC-Lab-vpc created in this lab.
4. Choose a route table to reflect the endpoint. Select the two private subnets as shown below. Additional routing information for using the endpoint is automatically added to the selected route table.

NOTE:- VPC endpoints are communications within the AWS network and have the security and compliance advantage of being able to control traffic through the endpoints. You can also optimize the data processing cost if you transfer your data through a VPC endpoint rather than a NAT gateway.

C.) Compute – Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2’s simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon’s proven computing environment.



The following hands-on are to be done .

* Launch web server instances and execute user data
* Set up a security group
* Create a custom Amazon Machine Image (AMI)
* Launch an Application Load Balancer (ALB)
* Configure a Launch Template
* Configure an Auto Scaling Group
* Test auto scaling and change manual settings

Launch instance and connect to web service

1. In the AWS console search bar, type EC2 and select it. Then click EC2 Dashboard at the top of the left menu. Press the Launch instance button and select Launch instance from the menu.
2. In Name, put the value Web server for custom AMI. And check the default setting in Amazon Machine Image below.
3. Select t2.micro in Instance Type.
4. For Key pair, choose Proceed without a key pair.
5. Click the Edit button in Network settings to set the space where EC2 will be located.
6. And choose the VPC-Lab-vpc created in the previous lab, and for the subnet, choose public subnet. Auto-assign public IP is set to Enable.

Create Security groups to act as a network firewall. Security groups will specify the protocols and addresses you want to allow in your firewall policy.

1. For the security group you are currently creating, this is the rule that applies to the EC2 that will be created. After entering Immersion Day - Web Server in Security group name and Description, select Add Security group rule and set HTTP to Type. Also allow TCP/80 for Web Service by specifying it. Select My IP in the source.
2. All other values accept the default values, expand by clicking on the Advanced Details tab at the bottom of the screen.
3. Click the Meta Data version dropdown and select V2 only (token required)
4. Wait for the instance's Instance state result to be Running. Open a new web browser tab and enter the Public DNS or IPv4 Public IP of your EC2 instance in the URL address field. If the page is displayed as shown below, the web server instance is configured normally.

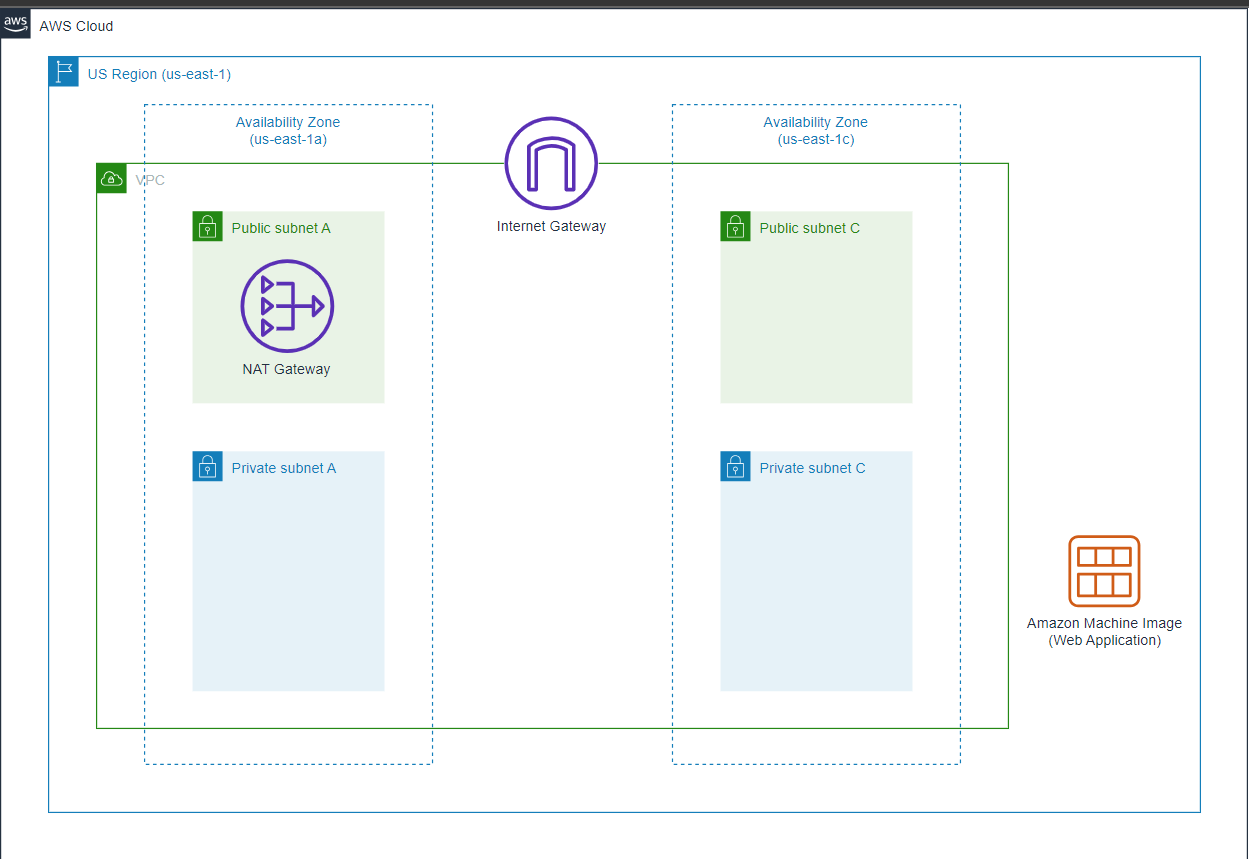
Create a custom AMI.

1. In the EC2 console, select the instance that we made earlier in this lab, and click Actions > Image and templates > Create Image.
2. In the Create Image console, type as shown below and press Create image to create the custom image.
4. Verify in the console that the image creation request in completed.
5. In the left navigation panel, Click the AMIs button located under IMAGES. You can see that the Status of the AMI that you just created. It will show either Pending or Available.

Terminate the instance

Custom AMI (Golden Image) creation has been completed for the auto scaling by using the EC2 instance you just created. Therefore, the EC2 instance currently running is no longer needed, so let's try to terminate it. ( In Deploy auto scaling web service, we will use custom AMI to create a new web server.)

NOTE - Custom AMI (Golden Image) creation has been completed for the auto scaling by using the EC2 instance you just created. Therefore, the EC2 instance currently running is no longer needed, so let's try to terminate it. ( In Deploy auto scaling web service, we will use custom AMI to create a new web server.)



Lauch Application Load Balancer

Using the network infrastructure created in the Network- Amazon VPC lab, we will deploy a web service that can automatically scale out/in under load and ensure high availability. We use the web server AMI created in the previous chapter and the network infrastructure named VPC-Lab.

Configure Application Load Balancer

AWS Elastic Load Balancer supports three types of load balancers: Application Load Balancer, Network Load Balancer, and Classic Load Balancer. In this lab, you will configure and set up the Application Load Balancer to handle load balancing HTTP requests.

1. From the EC2 Management Console in the left navigation panel, click Load Balancers under Load Balancing. Then click Create Load Balancer. In the Select load balancer type, click the Create button under Application Load Balancer.
2. Name the load balancer. In this case, name Name as Web-ALB. Leave the other settings at their default values.
3. Scrolling down a little bit, there is a section for selecting availability zones. First, Select the VPC-Lab-vpc created previously. For Availability Zones select the 2 public subnets that were created previously. This should be Public Subnet for ap-northeast-2a and Public Subnet C for us-east-1c.
4. In the Security groups section, click the Create new security group hyperlink. Enter web-ALB-SG as the security group name and check the VPC information. Click the Add rule button and select HTTP as the Type and Anywhere-IPv4 as the Source. And create a security group.
5. Return to the load balancer page again, click the refresh button, and select the web-ALB-SG you just created. Remove the default security group.
6. In Listeners and routing column, click Create target group. Put Web-TG for Target group name and check all settings same with the screen below. After that click Next button.
7. This is where we would register our instances. However, as we mentioned earlier, there are not instances to register at this moment. Click Create target group.
8. Again, move into the Load balancers page, click refresh button and select Web-TG. And then Click Create load balancer.

Configure Launch Template

Now that ALB has been created, it's time to place the instances behind the load balancer. To configure an Amazon EC2 instance to start with Auto Scaling Group, you can use **Launch Template**, **Launch Configuration**, or **EC2 Instance**. In this workshop, we will use the **Launch Template** to create an Auto Scaling group.

The launch template configures all parameters within a resource at once, reducing the number of steps required to create an instance. Launch templates make it easier to implement best practices with support for Auto Scaling and spot fleets, as well as spot and on-demand instances. This helps you manage costs more conveniently, improve security, and minimize the risk of deployment errors.

The launch template contains information that Amazon EC2 needs to start an instance, such as AMI and instance type. The Auto Scaling group refers to this and adds new instances when a scaling out event occurs. If you need to change the configuration of the EC2 instance to start in the Auto Scaling group, you can create a new version of the launch template and assign it to the Auto Scaling group. You can also select a specific version of the launch template that you use to start an EC2 instance in the Auto Scaling group, if necessary. You can change this setting at any time.

Create Security Group