**Create VPC**

Login to AWS management console and navigate to the VPC console. Select “Your VPCs” from sidebar and click on “Create VPC”

Graphical user interface, text, application, email

Description automatically generated

Enter VPC CIDR details as belowName tag: **MyVPC**  
IPv4 CIDR block\* : **10.0.0.0/16**

Graphical user interface, text, application

Description automatically generated

Leave the remaining settings as it is and click “Yes, Create” button.  Note that, upon creation of “MyVPC”, it will automatically create a default “Route table” and a “Network ACL” for “MyVPC”.

#### Create Public Subnet

Make sure to select “MyVPC”  under “VPC”  drop down menu and enter 10.0.1.0/24 in “IPv4 CIDR block” for public subnet.

Graphical user interface, application

Description automatically generated

#### Create Private Subnet

Now create Private subnet with CIDR 10.0.2.0/24

Graphical user interface, application

Description automatically generated

#### Add a route to Public Subnet

In order to make the instances created in Public subnet internet accessible, we will create a new “Route table”, that will establish connection between Public subnet and Internet Gateway.

Note: It is not recommended to modify “main” route table that was created by default when “MyVPC” was lanched.

Create a new Route table with name “PublicRT”

Select “PublicRT”, click on “Routes” tab and “edit”.

Graphical user interface, text, application, email

Description automatically generated

Add “0.0.0.0/0” as “Destination” and click on next box, it should list the “Internet Gateway” that we created before, select it and click on “Save”.

Graphical user interface, application

Description automatically generated

Associate this “PublicRT” with Public subnet “PublicNet”

Graphical user interface, text, application, email

Description automatically generated

Select  “PublicNet” and click “Save”

Graphical user interface, text, application, Word

Description automatically generated

see the following instructions to create one.

Graphical user interface, text, application

Description automatically generated

In the “Create Role” wizard, select **AWS service** role type and then select **EC2** service and **EC2** use case

Graphical user interface, application

Description automatically generated

Select an existing S3 access policy or click **Create policy** to define a new policy. If you are just getting started, you can select a built-in policy called “AmazonS3FullAccess”, which provides full access to S3 buckets that are part of your account:

Graphical user interface, text, application

Description automatically generated

When done attaching the policy, click **Next: Review**.

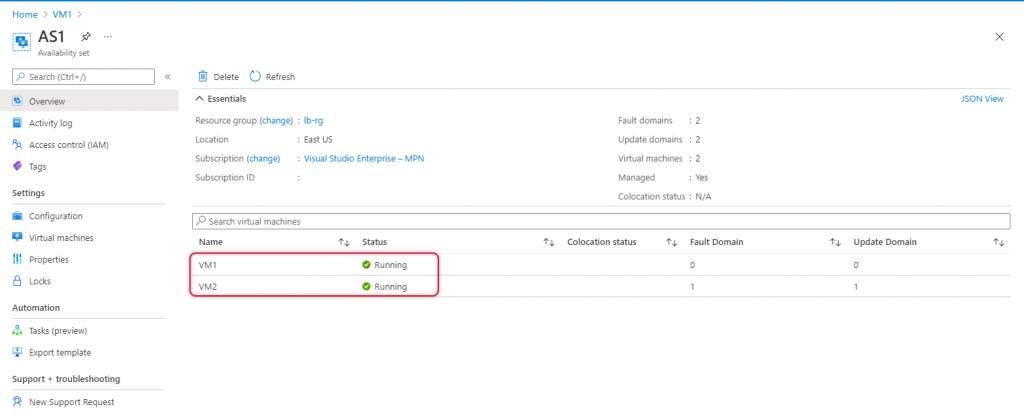
Graphical user interface, text, application, email

Description automatically generated

Click **Create role** to finish the role creation process.

**Create load balancer**

1. Frontend IP address.
2. Backend pool.
3. Inbound load-balancing rules.
4. Health probe.



A screenshot of a computer

Description automatically generated

**Send AWS Application Load Balancer Traffic to an EC2 Instance**

1. First, navigate to the EC2 Dashboard > Load Balancers > Select your ALB > Select 'Targets' tab > Select 'Edit'
2. Select the test server(s) you want to distribute traffic to and click 'Add to Registered', then click 'Save'
3. Choose **Create Auto Scaling group**.
4. In steps 1 and 2, choose the options as desired and proceed to **Step 3: Configure advanced options**.
5. For **Load balancing**, choose **Attach to a new load balancer**.
   1. Under **Attach to a new load balancer**, for **Load balancer type**, choose whether to create an Application Load Balancer or Network Load Balancer.
   2. For **Load balancer name**, enter a name for the load balancer, or keep the default name.
   3. For **Load balancer scheme**, choose whether to create a public internet-facing load balancer, or keep the default for an internal load balancer.
   4. For **Availability Zones and subnets**, select the public subnet for each Availability Zone in which you chose to launch your EC2 instances. (These prepopulate from step 2.).
   5. For **Listeners and routing**, update the port number for your listener (if necessary), and under **Default routing**, choose **Create a target group**. Alternatively, you can choose an existing target group from the drop-down list.
   6. If you chose **Create a target group** in the last step, for **New target group name**, enter a name for the target group, or keep the default name.
   7. To add tags to your load balancer, choose **Add tag**, and provide a tag key and value for each tag.
6. Proceed to create the Auto Scaling group. Your instances will be automatically registered to the load balancer after the Auto Scaling group has been created.
7. On the navigation pane, under **AUTO SCALING**, choose **Auto Scaling Groups**. Select the Auto Scaling group and verify that the maximum size of the Auto Scaling group is large enough that you can add another instance. Otherwise, on the **Details** tab, increase the maximum capacity.
8. On the navigation pane, under **Instances**, choose **Instances**, and then select an instance.
9. Choose **Actions**, **Instance settings**, **Attach to Auto Scaling Group**.
10. On the **Attach to Auto Scaling group** page, for **Auto Scaling Group**, select the Auto Scaling group, and then choose **Attach**.
11. If the instance doesn't meet the criteria, you get an error message with the details. For example, the instance might not be in the same Availability Zone as the Auto Scaling group. Choose **Close** and try again with an instance that meets the criteria.

## Attach an instance (AWS CLI)

**To attach an instance to an Auto Scaling group**

1. Describe a specific Auto Scaling group using the

aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg

The following example response shows that the desired capacity is 2 and that the group has two running instances.

{

"AutoScalingGroups": [

{

"AutoScalingGroupARN": "arn",

"ServiceLinkedRoleARN": "arn",

"TargetGroupARNs": [],

"SuspendedProcesses": [],

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"Tags": [],

"EnabledMetrics": [],

"LoadBalancerNames": [],

"AutoScalingGroupName": "my-asg",

"DefaultCooldown": 300,

"MinSize": 1,

"Instances": [

{

"ProtectedFromScaleIn": false,

"AvailabilityZone": "us-west-2a",

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"InstanceId": "i-05b4f7d5be44822a6",

"HealthStatus": "Healthy",

"LifecycleState": "Pending"

},

{

"ProtectedFromScaleIn": false,

"AvailabilityZone": "us-west-2a",

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"InstanceId": "i-0c20ac468fa3049e8",

"HealthStatus": "Healthy",

"LifecycleState": "InService"

}

],

"MaxSize": 5,

"VPCZoneIdentifier": "subnet-c87f2be0",

"HealthCheckGracePeriod": 300,

"TerminationPolicies": [

"Default"

],

"CreatedTime": "2019-03-18T23:30:42.611Z",

"AvailabilityZones": [

"us-west-2a"

],

"HealthCheckType": "EC2",

"NewInstancesProtectedFromScaleIn": false,

"DesiredCapacity": 2

}

]

}

aws autoscaling attach-instances --instance-ids i-0787762faf1c28619 --auto-scaling-group-name my-asg

The following example response shows that the desired capacity has increased by 1 instance (to a new capacity of 3), and that there is a new instance, i-0787762faf1c28619.

{

"AutoScalingGroups": [

{

"AutoScalingGroupARN": "arn",

"ServiceLinkedRoleARN": "arn",

"TargetGroupARNs": [],

"SuspendedProcesses": [],

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"Tags": [],

"EnabledMetrics": [],

"LoadBalancerNames": [],

"AutoScalingGroupName": "my-asg",

"DefaultCooldown": 300,

"MinSize": 1,

"Instances": [

{

"ProtectedFromScaleIn": false,

"AvailabilityZone": "us-west-2a",

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"InstanceId": "i-05b4f7d5be44822a6",

"HealthStatus": "Healthy",

"LifecycleState": "Pending"

},

{

"ProtectedFromScaleIn": false,

"AvailabilityZone": "us-west-2a",

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"InstanceId": "i-0c20ac468fa3049e8",

"HealthStatus": "Healthy",

"LifecycleState": "InService"

},

{

"ProtectedFromScaleIn": false,

"AvailabilityZone": "us-west-2a",

"LaunchTemplate": {

"LaunchTemplateName": "my-launch-template",

"Version": "1",

"LaunchTemplateId": "lt-050555ad16a3f9c7f"

},

"InstanceId": "i-0787762faf1c28619",

"HealthStatus": "Healthy",

"LifecycleState": "InService"

}

],

"MaxSize": 5,

"VPCZoneIdentifier": "subnet-c87f2be0",

"HealthCheckGracePeriod": 300,

"TerminationPolicies": [

"Default"

],

"CreatedTime": "2019-03-18T23:30:42.611Z",

"AvailabilityZones": [

"us-west-2a"

],

"HealthCheckType": "EC2",

"NewInstancesProtectedFromScaleIn": false,

"DesiredCapacity": 3

}

]

}