

## 1. Create one VPC in N. Virginia region.

Goto VPC in N. Virginia region and create, give CIDR range:

The screenshot shows the 'Create VPC' configuration page. The 'VPC settings' section is active, showing options for creating a VPC resource. A 'Name tag - optional' field contains 'my-vpc-01'. The 'IPv4 CIDR block' field is set to '192.168.0.0/24'. Under 'IPv6 CIDR block', the 'No IPv6 CIDR block' option is selected. In the 'Tenancy' section, 'Default' is chosen. The 'Tags' section shows a single tag named 'Name' with value 'my-vpc-01'.

VPC settings

Resources to create [Info](#)  
Create only the VPC resource or the VPC and other networking resources.

VPC only  VPC and more

Name tag - *optional*  
Creates a tag with a key of 'Name' and a value that you specify.  
my-vpc-01

IPv4 CIDR block [Info](#)  
 IPv4 CIDR manual input  IPAM-allocated IPv4 CIDR block  
192.168.0.0/24  
CIDR block size must be between /16 and /28.

IPv6 CIDR block [Info](#)  
 No IPv6 CIDR block  IPAM-allocated IPv6 CIDR block  Amazon-provided IPv6 CIDR block  IPv6 CIDR owned by me

Tenancy [Info](#)  
Default

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - <i>optional</i>
Q Name	X my-vpc-01

[Remove tag](#)

## 2. Create two subnets: one public subnet and one private subnet.

Create an Internet Gateway:

The screenshot shows the 'Internet gateways' list. There is one internet gateway entry: 'igw-0e85262989c8ff2d0' (Status: Attached, VPC ID: vpc-05f40aba9ace42ad, Owner: 471451201019). The 'Actions' menu has an option to 'Create internet gateway'.

Internet gateways (1) <a href="#">Info</a>					
<a href="#">Find internet gateways by attribute or tag</a>					
Name	Internet gateway ID	State	VPC ID	Owner	Actions
-	igw-0e85262989c8ff2d0	Attached	vpc-05f40aba9ace42ad   default-vpc	471451201019	<a href="#">Actions</a> <a href="#">Create internet gateway</a>

VPC > Internet gateways > Create internet gateway

### Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

**Internet gateway settings**

Name tag  
Creates a tag with a key of 'Name' and a value that you specify.

**Tags - optional**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-igw"/> <span style="color: red;">X</span> <span style="float: right;"><a href="#">Remove</a></span>

[Add new tag](#)  
You can add 49 more tags.

[Cancel](#) [Create internet gateway](#)

And attach to your vpc:

VPC dashboard < Attach to a VPC

Internet gateways (1/2) Info

The following internet gateway was created: igw-0e6f11033bd912ba4 - my-igw. You can now attach to a VPC to enable the VPC to communicate with the internet.

Name	Internet gateway ID	State	VPC ID	Owner
my-igw	igw-0e6f11033bd912ba4	Attached	vpc-05f40aba9ace42ad   default-vpc	471451201019

Actions [Create internet gateway](#)

The following internet gateway was created: igw-0e6f11033bd912ba4 - my-igw. You can now attach to a VPC to enable the VPC to communicate with the internet.

**Attach to VPC (igw-0e6f11033bd912ba4) Info**

**VPC**  
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

**Available VPCs**  
Attach the internet gateway to this VPC.

**AWS Command Line Interface command**

[Cancel](#) [Attach internet gateway](#)

Create Subnet:

VPC dashboard < Actions [Create subnet](#)

Subnets (6) Info

Last updated 7 minutes ago

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR	IPv6 CIDR	IPv6 CIDR association ID
-	subnet-0242f466ea951417e	Available	vpc-05f40aba9ace42ad   defa...	Off	172.31.16.0/20	-	-
-	subnet-09b641f51cb9a526	Available	vpc-05f40aba9ace42ad   defa...	Off	172.31.0.0/20	-	-
-	subnet-0d245a5e999877b45	Available	vpc-05f40aba9ace42ad   defa...	Off	172.31.80.0/20	-	-
-	subnet-040c366f14d04d758	Available	vpc-05f40aba9ace42ad   defa...	Off	172.31.64.0/20	-	-
-	subnet-0fb402a701cff6231	Available	vpc-05f40aba9ace42ad   defa...	Off	172.31.32.0/20	-	-
-	subnet-077bd529b150401c	Available	vpc-05f40aba9ace42ad   defa...	Off	172.31.48.0/20	-	-

Give the CIDR of subnet:

VPC > Subnets > Create subnet

### Create subnet Info

**VPC**

**VPC ID**  
Create Subnets in this VPC.  
vpc-047a3ca647314148b (my-vpc-01)

**Associated VPC CIDRs**

**IPv4 CIDRs**  
192.168.0.0/24

**Subnet settings**  
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

**Subnet name**  
Create a tag with a key of 'Name' and a value that you specify.  
pub-subnet  
The name can be up to 256 characters long.

**Availability Zone** Info  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.  
United States (Virginia) / us-east-1a (us-east-1a)

**IPv4 VPC CIDR block** Info  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.  
192.168.0.0/24

**IPv4 subnet CIDR block**  
192.168.0.0/28  
16 IPs

## Associate the Route Table with subnet:

Note: here I have used the Route Table which got created with the VPC creation:

VPC > Route tables

**Route tables (1/2) Info**

You have successfully updated subnet associations for rtb-054b02dcdd767b95 / pub-RT.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f59cb5fdfee	6 subnets	-	Yes	vpc-05f40ab9ace42ad   defa...	471451201019
<b>pub-RT</b>	rtb-054b02dcdd767b95	subnet-04b3b08ea0263cd4e...	-	Yes	vpc-047a3ca647314148b   my...	471451201019

**rtb-054b02dcdd767b95 / pub-RT**

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

**Explicit subnet associations (1)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
pub-subnet	subnet-04b3b08ea0263cd4e	192.168.0.0/28	-

## Add the Route i.e., IGW to the Route Table Route:

VPC > Route tables

**Route tables (1/2) Info**

You have successfully updated subnet associations for rtb-054b02dcdd767b95 / pub-RT.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f59cb5fdfee	6 subnets	-	Yes	vpc-05f40ab9ace42ad   defa...	471451201019
<b>pub-RT</b>	rtb-054b02dcdd767b95	subnet-04b3b08ea0263cd4e...	-	Yes	vpc-047a3ca647314148b   my...	471451201019

**rtb-054b02dcdd767b95 / pub-RT**

Details | **Routes** | Subnet associations | Edge associations | Route propagation | Tags

**Routes (1)**

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	Create Route Table

[VPC](#) > [Route tables](#) > [rtb-054b02dcde767b95](#) > Edit routes

### Edit routes

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	CreateRouteTable
Q_ 0.0.0.0/0	Internet Gateway	-	No	CreateRoute
	Q_ lgw-0e6f11033bd912ba4			

[Add route](#) [Cancel](#) [Preview](#) [Save changes](#)

[VPC dashboard](#) < [Route tables](#) > [rtb-054b02dcde767b95](#) [Actions](#)

Updated routes for rtb-054b02dcde767b95 / pub-RT successfully  
► Details

### rtb-054b02dcde767b95 / pub-RT

Details	Info	Explicit subnet associations	Edge associations
Route table ID <a href="#">rtb-054b02dcde767b95</a>	Main <input checked="" type="checkbox"/> Yes	subnet-04d3b08ea0265cd4e / pub-subnet	-
VPC <a href="#">vpc-047a3ca647314148b   my-vpc-01</a>	Owner ID <a href="#">471451201019</a>		

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

#### Routes (2)

Destination	Target	Status	Propagated	Route Origin
Q_ 0.0.0.0/0	lgw-0e6f11033bd912ba4	Active	No	Create Route
192.168.0.0/24	local	Active	No	Create Route Table

## Create another subnet (for private):

[VPC](#) > [Subnets](#) > Create subnet

### Create subnet [info](#)

**VPC**

**VPC ID**  
Create subnets in this VPC.  
[vpc-047a3ca647314148b \(my-vpc-01\)](#)

**Associated VPC CIDRs**

**IPv4 CIDRs**  
192.168.0.0/24

**Subnet settings**  
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

**Subnet name**  
Create a tag with a key of 'Name' and a value that you specify.  
  
The name can be up to 256 characters long.

**Availability Zone [info](#)**  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.  
[United States \(N. Virginia\) / us-east-1a \(us-east-1b\)](#)

**IPv4 VPC CIDR block [info](#)**  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 16 IPs

## Create Route Table:

VPC > Route tables > Create route table

### Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

<b>Route table settings</b>
Name - optional Create a tag with a key of 'Name' and a value that you specify. <input type="text" value="Pvt-RT"/>
VPC The VPC to use for this route table. <input type="text" value="vpc-047a3ca647314148b (my-vpc-01)"/>
<b>Tags</b> A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
Key <input type="text" value="Name"/> Value - optional <input type="text" value="pvt-RT"/> <a href="#">Remove</a>
<a href="#">Add new tag</a> You can add 49 more tags.

[Cancel](#) [Create route table](#)

## Associate Route Table with private Subnet:

VPC > Route tables

You have successfully updated subnet associations for rtb-0b9786d7a6f5b05f5 / pvt-RT.

Route tables (1/3) <small>Info</small>						
<a href="#">Find route tables by attribute or tag</a>						
Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f39cd85fdee	6 subnets	-	Yes	vpc-05f40aba9ace42ad   defa...	471451201019
pub-RT	rtb-054b02dcedd767b95	subnet-04b3b08ea0263c...	-	Yes	vpc-047a3ca647314148b   my...	471451201019
<b>pvt-RT</b>	<b>rtb-0b9786d7a6f5b05f5</b>	<b>subnet-0f6ef47e970ca02...</b>	-	No	vpc-047a3ca647314148b   my...	471451201019

rtb-0b9786d7a6f5b05f5 / pvt-RT

Subnet associations						
<a href="#">Edit subnet associations</a>						
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR			
pvt-subnet	subnet-0f6ef47e970ca0297	192.168.0.16/28	-			

Route Table does not contain any Route to IGW so it is private Route Table:

VPC > Route tables

You have successfully updated subnet associations for rtb-0b9786d7a6f5b05f5 / pvt-RT.

Route tables (1/3) <small>Info</small>						
<a href="#">Find route tables by attribute or tag</a>						
Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f39cd85fdee	6 subnets	-	Yes	vpc-05f40aba9ace42ad   defa...	471451201019
pub-RT	rtb-054b02dcedd767b95	subnet-04b3b08ea0263c...	-	Yes	vpc-047a3ca647314148b   my...	471451201019
<b>pvt-RT</b>	<b>rtb-0b9786d7a6f5b05f5</b>	<b>subnet-0f6ef47e970ca02...</b>	-	No	vpc-047a3ca647314148b   my...	471451201019

rtb-0b9786d7a6f5b05f5 / pvt-RT

Routes (1)						
<a href="#">Edit routes</a>						
Destination	Target	Status	Propagated			
192.168.0.0/24	local	Active	No			

-----done-----

## 3. Attach an IGW to the VPC.

### Create an Internet Gateway:

VPC > Internet gateways

You have successfully attached igw-0e832b2989cf8f2d0 to default-vpc.

Internet gateways (1) <small>Info</small>						
<a href="#">Find internet gateways by attribute or tag</a>						
Name	Internet gateway ID	State	VPC ID	Owner		
-	igw-0e832b2989cf8f2d0	Attached	vpc-05f40aba9ace42ad   default-vpc	471451201019		

VPC > Internet gateways > Create internet gateway

### Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

**Internet gateway settings**

Name tag  
Creates a tag with a key of 'Name' and a value that you specify.

**Tags - optional**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-igw"/> <span style="color: red;">X</span> <span style="float: right;"><a href="#">Remove</a></span>

[Add new tag](#)

You can add 49 more tags.

[Cancel](#) [Create internet gateway](#)

And attach to your vpc:

VPC dashboard < Internet gateways

The following internet gateway was created: igw-0e6f11033bd912ba4 - my-igw. You can now attach to a VPC to enable the VPC to communicate with the internet.

### Internet gateways (1/2) Info

Name	Internet gateway ID	State	VPC ID	Owner
my-igw	igw-0e6f11033bd912ba4	Attached	vpc-05f40aba9ace42ad   default-vpc	471451201019

The following internet gateway was created: igw-0e6f11033bd912ba4 - my-igw. You can now attach to a VPC to enable the VPC to communicate with the internet.

### Attach to VPC (igw-0e6f11033bd912ba4) Info

**VPC**  
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

**Available VPCs**  
Attach the internet gateway to this VPC.

**AWS Command Line Interface command**

[Cancel](#) [Attach internet gateway](#)

-----done-----

## 4. Create one public route table (RT) and one private route table.

Add the Route i.e., IGW to the Route Table Route:

VPC dashboard < Route tables

You have successfully updated subnet associations for rtb-054b02dcdd767b95 / pub-RT.

### Route tables (1/2) Info

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f3c0b5fdefee	5 subnets	-	Yes	vpc-05f40aba9ace42ad   def...	471451201019
pub-RT	rtb-054b02dcdd767b95	subnet-04b3b0flea0263c...	-	Yes	vpc-047a3ca647314148b   my...	471451201019

rtb-054b02dcdd767b95 / pub-RT

**Routes (1)**

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	CreateRouteTable

[Edit routes](#)

**Edit routes**

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	CreateRouteTable
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="Internet Gateway"/>	-	No	CreateRoute

[Add route](#)

[Cancel](#) [Preview](#) [Save changes](#)

VPC > Route tables > rtb-054b02dcde767b95

Updated routes for rtb-054b02dcde767b95 / pub-RT successfully

rtb-054b02dcde767b95 / pub-RT

**Details**

Route table ID rtb-054b02dcde767b95	Main Yes	Explicit subnet associations subnet-04d5b08ea0265c04e / pub-subnet	Edge associations -
VPC vpc-047a3ca647314148b   my-vpc-01	Owner ID 471451201019		

**Routes**

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	ipw-0cf110336e912ba4	Active	No	Create Route
192.168.0.0/24	local	Active	No	Create Route Table

## Create another subnet (for private):

VPC > Subnets > Create subnet

**Create subnet**

**VPC**

**VPC ID**  
Create subnets in this VPC.  
vpc-047a3ca647314148b (my-vpc-01)

**Associated VPC CIDRs**

**IPv4 CIDRs**  
192.168.0.0/24

**Subnet settings**  
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

**Subnet name**  
Create a tag with a key of 'Name' and a value that you specify.  
pvt-subnet  
The name can be up to 256 characters long.

**Availability Zone**  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.  
United States (N. Virginia) / us-east-1a (us-east-1b)

**IPv4 VPC CIDR block**  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.  
192.168.0.0/24

**IPv4 subnet CIDR block**  
192.168.0.16/28  
16 IPs

## Create Route Table:

VPC > Route tables > Create route table

**Create route table**

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

**Route table settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.  
pvt-RT

**VPC**  
The VPC to use for this route table.  
vpc-047a3ca647314148b (my-vpc-01)

**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key Q Name	Value - optional Q pvt-RT	Remove
---------------	------------------------------	--------

Add new tag  
You can add 49 more tags.

**Create route table**

## Associate Route Table with private Subnet:

[VPC](#) > [Route tables](#) > Create route table

**Create route table** Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

**Route table settings**

Name - optional  
Create a tag with a key of 'Name' and a value that you specify.

VPC  
The VPC to use for this route table.

**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Q Name"/>	<input type="text" value="Q pvt-RT"/> <span style="color: red;">X</span>
<a href="#">Add new tag</a>	

You can add 49 more tags.

[Cancel](#) [Create route table](#)

[VPC](#) > [Route tables](#)

**VPC dashboard** <

**Route tables** (1/3) Info

You have successfully updated subnet associations for rtb-0b9786d7a6f5b05f5 / pvt-RT.

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f9c0b5f5fdee	6 subnets	-	Yes	vpc-05f40ab9ace42ad   def...	471451201019
pub-RT	rtb-054b02dced767b95	subnet-04b3b0ea0263c...	-	Yes	vpc-047a5ca647314148b   my...	471451201019
<b>pvt-RT</b>	<b>rtb-0b9786d7a6f5b05f5</b>	<b>subnet-0f6e147e970ca02...</b>	-	No	<b>vpc-047a5ca647314148b   my...</b>	<b>471451201019</b>

**rtb-0b9786d7a6f5b05f5 / pvt-RT**

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

**Explicit subnet associations (1)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
pvt-subnet	subnet-0f6e147e970ca0297	192.168.0.16/28	-

[Edit subnet associations](#)

Route Table does not contain any Route to IGW so it is private Route Table:

[VPC](#) > [Route tables](#)

**VPC dashboard** <

**Route tables** (1/3) Info

You have successfully updated subnet associations for rtb-0b9786d7a6f5b05f5 / pvt-RT.

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC	Owner ID
default-vpc-RT	rtb-01567f9c0b5f5fdee	6 subnets	-	Yes	vpc-05f40ab9ace42ad   def...	471451201019
pub-RT	rtb-054b02dced767b95	subnet-04b3b0ea0263c...	-	Yes	vpc-047a5ca647314148b   my...	471451201019
<b>pvt-RT</b>	<b>rtb-0b9786d7a6f5b05f5</b>	<b>subnet-0f6e147e970ca02...</b>	-	No	<b>vpc-047a5ca647314148b   my...</b>	<b>471451201019</b>

**rtb-0b9786d7a6f5b05f5 / pvt-RT**

Details | **Routes** | Subnet associations | Edge associations | Route propagation | Tags

**Routes (1)**

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	Create Route Table

[Both](#) [Edit routes](#)

-----done-----

- Deploy a NAT gateway in the public subnet and attach the NAT gateway to the private subnet.

Create NAT Gateway:

The screenshot shows the AWS VPC NAT gateways dashboard. On the left, there's a sidebar with navigation links for VPC dashboard, Virtual private cloud, Security, and PrivateLink and Lattice. The main area is titled "NAT gateways info" and has a search bar. A table header includes columns for Name, NAT gateway ID, Connectivity..., State, State message, Primary public IP..., Primary private IP..., Primary network interface ID, and VPC. Below the table, a message says "No NAT gateways found".

Select public subnet and allocate the elastic ip:

The screenshot shows the "Create NAT gateway" wizard. It starts with a success message: "Elastic IP address 52.22.255.112 (eipalloc-034fe7574f76f201a) allocated." The "NAT gateway settings" step shows a "Name - optional" field with "my-ngw" and a "Subnet" dropdown set to "subnet-04b3b08ea0263cd4e (pub-subnet)". Under "Connectivity type", "Public" is selected. In the "Elastic IP allocation ID" step, "eipalloc-034fe7574f76f201a" is assigned. The "Tags" step shows a key-value pair "Name" with "my-ngw".

The screenshot shows the "NAT gateways (1/1)" details page. It lists one entry: "my-ngw" with NAT gateway ID "nat-0748f7f278466a59c", connectivity type "Public", state "Available", primary public IP "52.22.255.112", primary private IP "192.168.0.10", and VPC "vpc-047a3ca647314148b / my-vpc-01". The "nat-0748f7f278466a59c / my-ngw" details page shows the same information and includes tabs for Details, Secondary IPv4 addresses, Monitoring, and Tags.

Goto Route Table, select your private Route Table and Edit Routes:

Route tables (1/3) info

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
pub-RT	rtb-054b02dcedd767b95	subnet-04b3b0bea0263c...	-	Yes	vpc-047a3ca647314148b   my...	471451201019
<b>pvt-RT</b>	<b>rtb-0b9786d7a6f5b05f5</b>	<b>subnet-0f6ef47e970ca02...</b>	-	No	vpc-047a3ca647314148b   my...	471451201019
default-vpc-RT	rtb-01567f59cb5f1feee	6 subnets	-	Yes	vpc-05f40aba9acea42ad   defa...	471451201019

rtb-0b9786d7a6f5b05f5 / pvt-RT

Routes (1)

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	Create Route Table

Add the NAT gateway in the Routes:

Edit routes

Destination	Target	Status	Propagated	Route Origin
192.168.0.0/24	local	Active	No	CreateRouteTable
0.0.0.0/0	NAT Gateway	-	No	CreateRoute
	nat-0748f71278466a59c	-		

Add route

Route tables (1/3) info

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC	Owner ID
pub-RT	rtb-054b02dcedd767b95	subnet-04b3b0bea0263c...	-	Yes	vpc-047a3ca647314148b   my...	471451201019
<b>pvt-RT</b>	<b>rtb-0b9786d7a6f5b05f5</b>	<b>subnet-0f6ef47e970ca02...</b>	-	No	vpc-047a3ca647314148b   my...	471451201019
default-vpc-RT	rtb-01567f59cb5f1feee	6 subnets	-	Yes	vpc-05f40aba9acea42ad   defa...	471451201019

rtb-0b9786d7a6f5b05f5 / pvt-RT

Routes (2)

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	nat-0748f71278466a59c	Active	No	Create Route
192.168.0.0/24	local	Active	No	Create Route Table

-----done-----

## 6. Create two instances, one in the public subnet and one in the private subnet.

Launching the instance in public subnet, select your vpc and select your public subnet:

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The 'Network settings' section is expanded, showing a VPC (my-vpc-01) and a private subnet (subnet-04b3b08ea0263cd4). The 'Auto-assign public IP' dropdown is set to 'Disable'. The 'Summary' panel on the right indicates 1 instance will be launched with the AMI 'Amazon Linux 2023 AMI 2023.9.2...' and instance type 't3.micro'.

Launch another instance in private subnet by selecting your vpc and private subnet and ‘Disable’ the Auto-assign public IP:

This screenshot shows the same 'Launch an instance' wizard, but the 'Auto-assign public IP' dropdown is now set to 'Disable'. All other settings, including the VPC and subnet, remain the same as the previous screenshot.

-----done-----

## 7. Deploy Apache server on both EC2 instances with a sample index.html file.

Installing Apache on public EC2:

```

=====
Installing:
httpd          x86_64  2.4.65-1.amzn2023.0.1      amazonlinux   47 k
Installing dependencies:
apr            x86_64  1.7.5-1.amzn2023.0.4      amazonlinux   129 k
apr-util        x86_64  1.6.3-1.amzn2023.0.1      amazonlinux   98 k
generic-logos-httd noarch  18.0.0-12.amzn2023.0.3    amazonlinux   19 k
httpd-core       x86_64  2.4.65-1.amzn2023.0.1      amazonlinux   1.4 M
httpd-filesystem noarch  2.4.65-1.amzn2023.0.1      amazonlinux   13 k
httpd-tools      x86_64  2.4.65-1.amzn2023.0.1      amazonlinux   81 k
libbrotli       x86_64  1.0.9-4.amzn2023.0.2      amazonlinux   315 k
mailcap         noarch  2.1.49-3.amzn2023.0.3      amazonlinux   33 k
Installing weak dependencies:
apr-util-openssl x86_64  1.6.3-1.amzn2023.0.1      amazonlinux   17 k
mod_http2        x86_64  2.0.27-1.amzn2023.0.3      amazonlinux   166 k
mod_lua          x86_64  2.4.65-1.amzn2023.0.1      amazonlinux   60 k
=====
Transaction Summary
=====
Install 12 Packages
=====
```

[httpd enable and start:](#)

```
[ec2-user@ip-192-168-0-7 ~]$ vim /var/www/html/index.html
[ec2-user@ip-192-168-0-7 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr
/lib/systemd/system/httpd.service.
[ec2-user@ip-192-168-0-7 ~]$ sudo systemctl start httpd
```

[Deploy a sample index.html in /var/www/html/index.html:](#)

```
[ec2-user@ip-192-168-0-7 ~]$ cat /var/www/html/index.html
<h1>Techie Horizon</h1>
```

[And re-start httpd:](#)

```
[ec2-user@ip-192-168-0-7 ~]$ sudo systemctl restart httpd
```

[Then login to private Ec2 from Bastion host and install httpd:](#)

```
[ec2-user@ip-192-168-0-23 ~]$ sudo yum install httpd
Amazon Linux 2023 Kernel Livepatch repository 226 kB/s | 26 kB 00:00
Dependencies resolved.
=====
 Package          Arch    Version           Repository      Size
=====
Installing:
httpd            x86_64  2.4.65-1.amzn2023.0.1  amazonlinux   47 k
Installing dependencies:
apr              x86_64  1.7.5-1.amzn2023.0.4  amazonlinux   129 k
apr-util         x86_64  1.6.3-1.amzn2023.0.1  amazonlinux   98 k
generic-logos-httdp noarch  18.0.0-12.amzn2023.0.3  amazonlinux   19 k
httpd-core       x86_64  2.4.65-1.amzn2023.0.1  amazonlinux   1.4 M
httpd-filesystem noarch  2.4.65-1.amzn2023.0.1  amazonlinux   13 k
httpd-tools       x86_64  2.4.65-1.amzn2023.0.1  amazonlinux   81 k
libbrotli        x86_64  1.0.9-4.amzn2023.0.2  amazonlinux   315 k
mailcap          noarch  2.1.49-3.amzn2023.0.3  amazonlinux   33 k
Installing weak dependencies:
apr-util-openssl x86_64  1.6.3-1.amzn2023.0.1  amazonlinux   17 k
mod_http2        x86_64  2.0.27-1.amzn2023.0.3  amazonlinux   166 k
mod_lua          x86_64  2.4.65-1.amzn2023.0.1  amazonlinux   60 k
=====
Transaction Summary
```

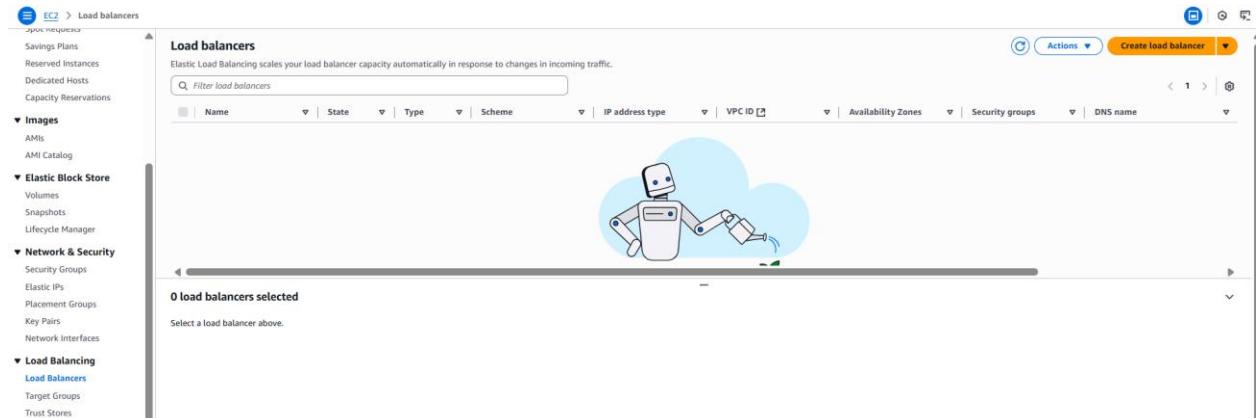
And enable, start the httpd service and deploy sample index.html and restart the httpd:

```
[ec2-user@ip-192-168-0-23 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-192-168-0-23 ~]$ sudo systemctl start httpd
[ec2-user@ip-192-168-0-23 ~]$ vim /var/www/html/index.html
[ec2-user@ip-192-168-0-23 ~]$ sudo vim /var/www/html/index.html
[ec2-user@ip-192-168-0-23 ~]$ sudo systemctl restart httpd
[ec2-user@ip-192-168-0-23 ~]$ sudo curl localhost:80
<h1>Techie Horizon from Privat Subnet</h1>
```

-----done-----

## 8. Create one application load balancer and attach it to both EC2 instances.

Create Load Balancer:



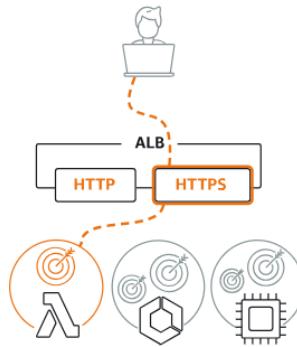
Create Application load Balancer:

## Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

### Load balancer types

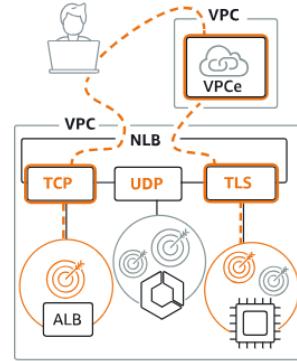
#### Application Load Balancer [Info](#)



Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

[Create](#)

#### Network Load Balancer [Info](#)



Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

[Create](#)

#### Gateway Load Balancer [Info](#)



Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.

[Create](#)

## Give the name and Scheme as Internet facing:

### Create Application Load Balancer [Info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

#### ▶ How Application Load Balancers work

#### Basic configuration

##### Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

s-apach1

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

##### Scheme [Info](#)

Scheme can't be changed after the load balancer is created.

**Internet-facing**

- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name resolves to public IPs.
- Requires a public subnet.

Internal

- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.
- Compatible with the IPv4 and Dualstack IP address types.

##### Load balancer IP address type [Info](#)

Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

IPv4

Includes only IPv4 addresses.

Dualstack

Includes IPv4 and IPv6 addresses.

Dualstack without public IPv4

Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with Internet-facing load balancers only.

## Select your vpc and at least 2 Availability zones:

**Network mapping** [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

**VPC** | [Info](#)

The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view [target groups](#).

vpc-0473ca647314148b (my-vpc-01) 192.168.0.24 [Create VPC](#)

**IP pools** | [Info](#)

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view [Pools](#) in the [Amazon VPC IP Address Manager console](#).

Use IPAM pool for public IPv4 addresses

The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

**Availability Zones and subnets** | [Info](#)

Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

us-east-1a (use1-az2)

**Subnet**  
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-04b3b08ea0263cd4e  
IPv4 subnet CIDR: 192.168.0.0/28 [pub-subnet](#)

us-east-1b (use1-az4)

**Subnet**  
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.

subnet-0f6ef47e970ca0297  
IPv4 subnet CIDR: 192.168.0.16/28 [pvt-subnet](#)

⚠ The selected subnet does not have a route to an internet gateway. This means that your load balancer will not receive internet traffic.  
You can proceed with this selection; however, for internet traffic to reach your load balancer, you must update the subnet's route table in the [VPC console](#).

## Give the Security Group:

**Security groups** [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**Security groups**

Select up to 5 security groups

default sg-09b0cccd13a4ab2ab0 VPC: vpc-0473ca647314148b

## Select protocol and port and click on Create Target group:

**Listeners and routing** [Info](#)

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

**Protocol** [HTTP](#) **Port** [80](#) Remove

**Default action** [Info](#)

The default action is used if no other rules apply. Choose the default action for traffic on this listener.

**Routing action**

Forward to target group  Redirect to URL  Return fixed response

**Forward to target group** [Info](#)

Choose a target group and specify routing weight or [create target group](#).

**Target group** Select a target group [Create](#) Weight 1 Percent 100% 0-999

**+ Add target group**  
You can add up to 4 more target groups.

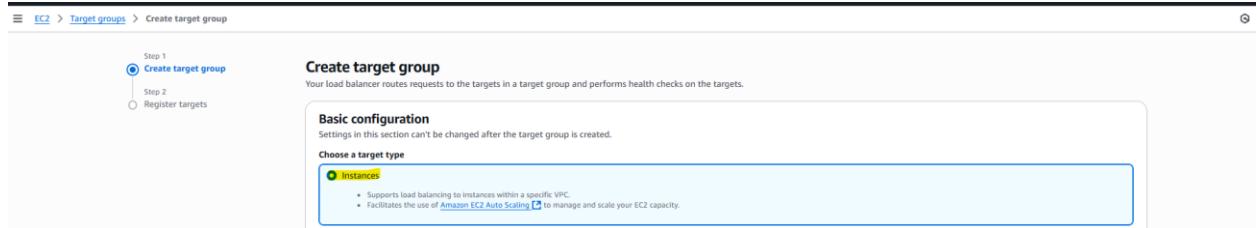
**Target group stickiness** [Info](#)

Enables the load balancer to bind a user's session to a specific target group. To use stickiness the client must support cookies. If you want to bind a user's session to a specific target, turn on the Target Group attribute Stickiness.

Turn on target group stickiness

**Listener tags - optional**  
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

## Select Instances:



And give Target Group name, Protocol Port , IPV4 and your VPC:

**Target group name**

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Protocol**

Protocol for load balancer-to-target communication. Can't be modified after creation.

HTTP
▼

**Port**

Port number where targets receive traffic. Can be overridden for individual targets during registration.

80
1-65535

**IP address type**

Only targets with the indicated IP address type can be registered to this target group.

IPv4

Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

IPv6

Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#) ▼

**VPC**

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

vpc-047a3ca647314148b (my-vpc-01)
▼
[Create VPC](#) ▼

**Protocol version**

HTTP1

Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

HTTP2

Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

gRPC

Send requests to targets using gRPC. Supported when the request protocol is gRPC.

And click on Next,

**Health checks**

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

**Health check protocol**

HTTP

**Health check path**

Use the default path of "/" to perform health checks on the root, or specify a custom path if preferred.

/

Up to 1024 characters allowed.

► Advanced health check settings

**Attributes**

① Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

► Tags - optional

Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them.

[Cancel](#) [Next](#)

Select your instances and 'Include as pending below':

Step 1  
Create target group  
Step 2  
Register targets

**Register targets**

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

**Available instances (2/2)**

Instance ID	Name	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-0caea086cc722d211	pvt-instance	Running	default	us-east-1b	192.168.0.23	subnet-046ef47e970ca02
i-07069417c26cb8a94	pub-instance	Running	default	us-east-1a	192.168.0.7	subnet-04b3b08ea0265cc

2 selected

Ports for the selected instances  
Ports for routing traffic to the selected instances.  
80  
1-65535 (separate multiple ports with commas)

[Include as pending below](#)

Then both the instances will get added to the Target group:

**EC2** > **Target groups** > apache-tg

**Details**

arn:aws:elasticloadbalancing:us-east-1:171451201019:targetgroup/apache-tg/a4c5e6ed961cedbc

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-047a3ca647314148b
IP address type	Load balancer		

2 Total targets      ② Healthy      ① Unhealthy      0 Unused      ① Initial      ① Draining

0 Anomalous

► Distribution of targets by Availability Zone (AZ)  
Select values in this table to see corresponding filters applied to the Registered targets table below.

**Targets** **Monitoring** **Health checks** **Attributes** **Tags**

**Registered targets (2) info**

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Instance ID	Name	Port	Zone	Health status	Health status details	Administrative state	Override details	Launch time	Anomaly detection
i-0caea086cc722d211	pvt-instance	80	us-east-1b (us...)	Healthy	-	① No override	No override is cu...	October 6, 2025,...	Normal
i-07069417c26cb8a94	pub-instance	80	us-east-1a (us...)	Healthy	-	① No override	No override is cu...	October 6, 2025,...	Normal

-----done-----

## 9. Store application load balancer logs in S3.

Storing **Application Load Balancer (ALB)** access logs in **Amazon S3** is a great way to monitor traffic, troubleshoot issues, and analyze performance.

### What You'll Need

- An **Application Load Balancer** already created.
- An **S3 bucket** (or permission to create one).
- IAM permissions to modify the ALB and S3 settings.

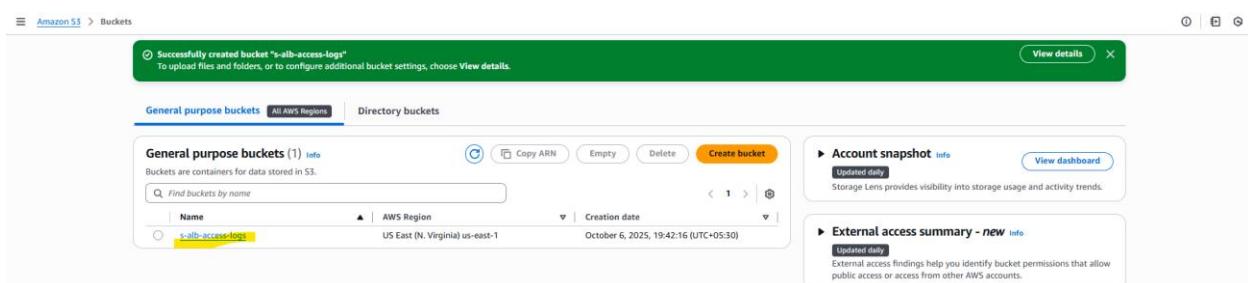
### Create or Use an S3 Bucket

You need an S3 bucket where the ALB can store logs.

1. Go to **Amazon S3 Console**.
2. Click "Create bucket".
3. Name the bucket (e.g., alb-access-logs-myapp).
4. Choose a region — ideally, the **same region** as your ALB.
5. In **Bucket settings for Block Public Access**, leave all options checked (recommended).
6. Click **Create bucket**.

**Note:** ALB logs are stored in the format:

AWSLogs/<account-id>/elasticloadbalancing/<region>/<year>/<month>/<day>/...



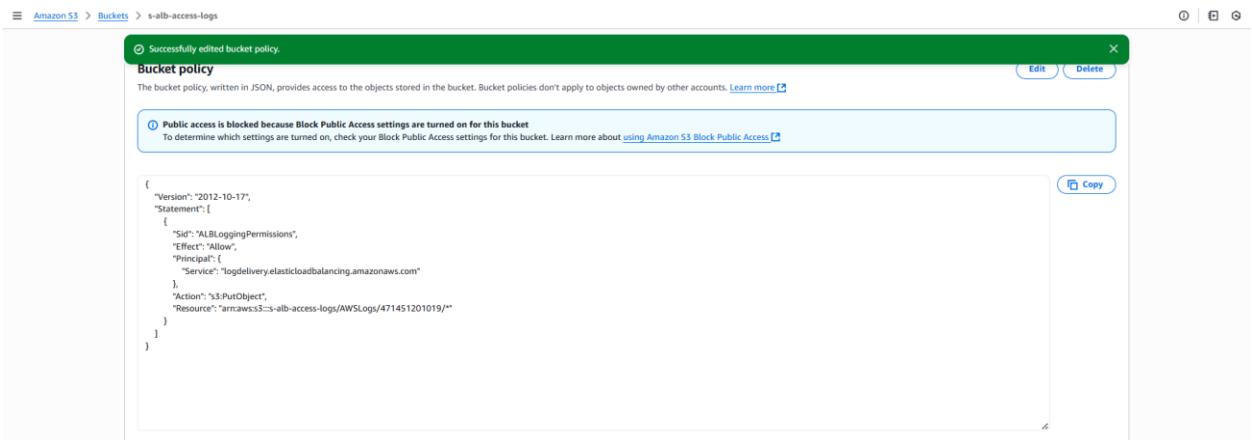
## 2: Allow ALB to Write Logs to S3

The ALB writes logs to your bucket using the **AWS Log Delivery** service principal. You need to add a **bucket policy** to allow this.

1. Go to your newly created S3 bucket.
2. Click **Permissions > Bucket policy**.

3. Add the following policy (replace placeholders with your values):

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "ALBLoggingPermissions",  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "logdelivery.elasticloadbalancing.amazonaws.com"  
      },  
      "Action": "s3:PutObject",  
      "Resource": "arn:aws:s3:::alb-access-logs-myapp/AWSLogs/<account-id>/*"  
    }  
]  
}
```



Note: This allows the ALB to write log files to your bucket.

3. Enable Access Logging on the ALB:

Go to the EC2 Console > Load Balancers.

Select your Application Load Balancer.

Click the "Description" tab.

Choose "Edit attributes".

Under Access logs:

- **Check the box:** Enable access logs.
- **S3 location:** Enter your bucket name (e.g., alb-access-logs-myapp)
- **Prefix (optional):** Add a folder name, like logs/ to organize files better.

Click Save.

The screenshot shows the AWS Elastic Load Balancer (ELB) configuration interface. The left sidebar lists various AWS services like EC2, Lambda, and CloudWatch. The main page displays a load balancer named 's-apache'. The 'Attributes' tab is active, showing traffic configuration details such as TLS version, WAF fail open, HTTP/2, and connection idle timeout. The 'Monitoring' tab shows the 'Access logs' section, where the 'Enable access logs' checkbox is checked and the 'S3 location' dropdown is set to 's-alb-access-logs'. A modal window titled 'Choose an S3 bucket' is open, listing a single bucket named 's-alb-access-logs'.

And save:

Your ALB is now configured to log requests into the S3 bucket.

Check the logs in s3:

-----done-----

## 10. Store the VPC flow logs in a CloudWatch log group.

Select your required vpc for which you want to create flow logs and click on create flow log:

Your VPCs (1/2) Info

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table
<input checked="" type="checkbox"/> my-vpc	vpc-0b28d6d65605ed45b	Available	Off	10.0.0.0/24	-	dopt-006bd7841ca77fa3	rtb-028b04470d161ec0d
<input type="checkbox"/> default-vpc	vpc-05140uba9ace42ad	Available	Off	172.31.0.0/16	-	dopt-006bd7841ca77fa3	rtb-01567f59c0b5f5feee

vpc-0b28d6d65605ed45b / my-vpc

Details | Resource map | CIDRs | **Flow logs** | Tags | Integrations

Flow logs (1)

Create Flow log

## Give the name , filter, maxm aggregation interval and Destination:

VPC > Your VPCs > Create flow logs

### Create flow log

Flow logs can capture IP traffic flow information for the network interfaces associated with your resources. You can create multiple flow logs to send traffic to different destinations.

**Selected resources**

Flow logs will only be created for resources in an available state.

Name	Resource ID	State
my-vpc	vpc-0b28d6d65605ed45b	Available

**Flow log settings**

**Name - optional**

vpcflowlog-cloudWatch

**Filter**

The type of traffic to capture (accepted traffic only, rejected traffic only, or all traffic).

Accept  
 Reject  
 All

**Maximum aggregation interval**

The maximum interval of time during which a flow of packets is captured and aggregated into a flow log record.

10 minutes  
 1 minute

**Destination**

The destination to which to publish the flow log data.

Send to CloudWatch Logs  
 Send to an Amazon S3 bucket  
 Send to Amazon Data Firehose in the same account  
 Send to Amazon Data Firehose in a different account

As we are storing in the cloudwatch log group so first we have to create the Log groupin CloudWatch, goto CloudWatch and select log group:

CloudWatch

CloudWatch

Overview

Get started with CloudWatch

Set alarms on any of your metrics to

Create and name any CloudWatch

Monitor using your existing system,

Write rules to indicate which events

Create log group:

The screenshot shows the CloudWatch Log groups page. The left sidebar includes sections for Dashboards, AI Operations, Alarms, Logs (with Log groups selected), Metrics, Application Signals (APM), Network Monitoring, and Insights. The main content area displays a table titled 'Log groups (0)' with a note: 'By default, we only load up to 10000 log groups.' A search bar at the top right says 'Filter log groups or try pattern search' with 'Exact match' checked. The table has columns for Log group, Log class, Anomaly d..., Data protection, Sensitive data count, Retention, Metric filters, and Contributor Insights. A message at the bottom states 'No log groups' and 'You have not created any log groups.' with a 'Create log group' button.

Provide the details and create log group:

The screenshot shows the 'Create log group' wizard. The left sidebar is identical to the previous screenshot. The main area is titled 'Create log group' and contains a 'Log group details' section. It includes a note about CloudWatch Logs log classes, a 'Log group name' input field containing 'vpc\_flowlog\_loggroup', a 'Retention setting' dropdown set to 'Never expire', a 'Log class' dropdown set to 'Standard', and a 'KMS key ARN - optional' input field. Below this is a 'Tags' section with a note about tags, a 'Add new tag' button, and a note about adding up to 50 tags. At the bottom are 'Cancel' and 'Create' buttons.

Then create the IAM role with policy for vpc flow log to store logs in CloudWatch:

IAM Policy:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Action": [  
                "logs:CreateLogGroup",  
                "logs:CreateLogStream",  
                "logs:PutLogEvents",  
                "logs:DescribeLogGroups",  
                "logs:DescribeLogStreams"  
            ],  
            "Effect": "Allow",  
            "Resource": "*"  
        }  
    ]  
}
```

And create the role with Trust policy:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": "vpc-flow-logs.amazonaws.com"  
            },  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

And assign the above created policy to this role and select this role while creating vpc flow log:

VPC > Your VPCs > Create flow logs

The maximum interval of time during which a flow of packets is captured and aggregated into a flow log record.

10 minutes  
 1 minute

**Destination**  
The destination to which to publish the flow log data.

Send to CloudWatch Logs  
 Send to an Amazon S3 bucket  
 Send to Amazon Data Firehose in the same account  
 Send to Amazon Data Firehose in a different account

**Destination log group** | Info  
The name of an existing log group or the name of a new log group that will be created when you create this flow log. A new log stream is created for each monitored network interface.

vpc-flowlog-loggroup (C)

[View this log group in the CloudWatch console](#) (C)

**Service access**  
VPC flow logs require permissions to create log groups and publish events in CloudWatch.

Use an existing service role  
 Create and use a new service role

**Service role** | Info  
The IAM role that has permission to publish to the Amazon CloudWatch log group.

vpc\_flowlog\_role (C)

[View this service role in the IAM console](#) (C)

**Log record format**  
Specify the fields to include in the flow log record.

AWS default format  
 Custom format

**Additional metadata**  
Include additional metadata to AWS default log record format.

Include Amazon ECS metadata

And the logs will get saved in the cloudWatch log group:

CloudWatch > Log groups > vpc-flowlog-loggroup > eni-0bd7de626ff7356bd2-all

**CloudWatch** <

Favorites and recents ▶

Dashboards New

▶ AI Operations New

▶ Alarms 0 ○ ○ ○

▼ Logs

- Log groups
- Log Anomalies
- Live Tail
- Logs Insights
- Contributor Insights

▶ Metrics

▶ Application Signals (APM) New

- GenAI Observability Preview

▶ Network Monitoring

▶ Insights

**Log events**  
You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#) (C)

Filter events - press enter to search (C)

Actions Start tailing Create metric filter

Timestamp	Message
No older events at this moment. <a href="#">Retry</a>	
2025-09-25T06:59:23.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783563 1758783594 - NODATA
	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783563 1758783594 - NODATA
2025-09-25T07:00:36.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 10.0.0.11 54.81.127.33 47572 123 17 1 76 1758783636 1758783637 ACCEPT OK
	2 4714512081819 eni-0bd7de626ff7356bd2 10.0.0.11 54.81.127.33 47572 123 17 1 76 1758783636 1758783637 ACCEPT OK
2025-09-25T07:00:59.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 10.0.0.11 3.87.127.143 38896 123 17 1 76 1758783689 1758783688 ACCEPT OK
	2 4714512081819 eni-0bd7de626ff7356bd2 10.0.0.11 13.218.199.211 52848 123 17 1 76 1758783689 1758783688 ACCEPT OK
2025-09-25T07:01:13.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783683 1758783714 - NODATA
	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783683 1758783714 - NODATA
2025-09-25T07:01:23.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783683 1758783834 - NODATA
	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783683 1758783834 - NODATA
2025-09-25T07:04:23.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 - - - - - 1758783683 1758783897 ACCEPT OK
	2 4714512081819 eni-0bd7de626ff7356bd2 10.0.0.11 13.220.37.97 32774 443 6 3 100 1758783897 1758783897 ACCEPT OK
2025-09-25T07:05:34.000Z	2 4714512081819 eni-0bd7de626ff7356bd2 10.0.0.11 13.220.37.97 54464 443 6 4 240 1758783934 1758783935 ACCEPT OK

**done**

## 11. Create monitoring dashboards to monitor CPU utilization and to monitor the Apache service.

Create ec2 instance

Create Dashboard in CloudWatch and provide the instance id and select pre-instances and select the cpuUtilization:



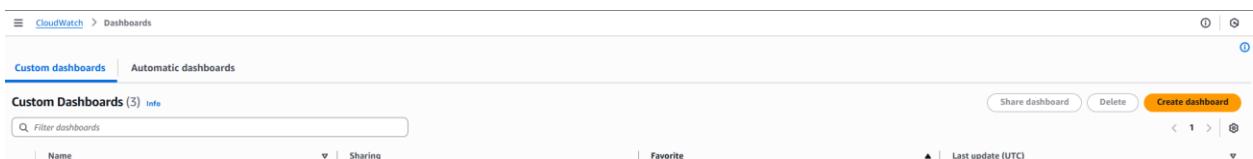
Install tomcat and all its dependencies.

Create a script to check the status of Tomcat:

```
[root@ip-172-31-30-63 ~]# cat monitor.sh
#!/bin/bash
# Check if Tomcat service is active
if systemctl is-active --quiet tomcat; then
    STATUS=1
else
    STATUS=0
fi
# Push custom metric to Cloudwatch
aws cloudwatch put-metric-data \
--namespace "TomcatMonitoring" \
--metric-name "TomcatStatus" \
--value $STATUS \
--region us-east-1
```

Run the script

And create the dashboard in CloudWatch:



**Create new dashboard**

**Dashboard name**

Valid characters in dashboard names include "0-9A-Za-z-\_".

**Cancel** **Create dashboard**

Select your required widget:

**Add widget**

**Data sources types**

- Cloudwatch
- Other content types
- Create data sources

**Widget Configuration**

**Data type**

- Metrics**
- Logs
- Alarms

**Widget type**

- Line**  
Compare metrics over time 
- Data table**  
Compare metrics values over time in a table 
- Number**  
Instantly see the latest value for a metric **75 %** 
- Gauge**  
See the latest value of a metric within a range 
- Stacked area**  
Compare the total over time 
- Bar**  
Compare categories of data 
- Pie**  
Show percentage or proportional data 
- Explorer**  
A single widget with multiple tag-based graphs 

**Cancel** **Next**

Select here the custom namespace i.e., tomcat:

**Add metric graph**

TomcatStatus

1h 3h 12h 1d 3d 1w Custom (1m) UTC timezone Number 10 seconds

Your CloudWatch graph is empty.  
Select some metrics to appear here.

Browse (1,385) Multi source query Graphed metrics Options Source Add math Add query

N. Virginia

Search for any metric, dimension, resource id or account id

Custom namespaces

- Custom/Nginx 1 TomcatMonitoring 1

AWS namespaces

- Bedrock/DataAutomation 5 EBS 247 EC2 380 Firehose 2
- Location 8 Logs 13 NATGateway 32 S3 2
- SNS 2 States 4 TransitGateway 180 Usage 508

Cancel Create widget

**Add metric graph**

TomcatStatus

1h 3h 12h 1d 3d 1w Custom (1m) UTC timezone Number 10 seconds

Your CloudWatch graph is empty.  
Select some metrics to appear here.

Browse (1) Multi source query Graphed metrics Options Source Add math Add query

All > TomcatMonitoring N. Virginia

Search for any metric, dimension, resource id or account id

Metrics with no dimensions 1

**Add metric graph**

TomcatStatus

Persist time range 1h 3h 12h 1d 3d 1w Custom (1m) UTC timezone Number 10 seconds

0

Browse (1) Multi source query Graphed metrics (1) Options Source Add math Add query

All > TomcatMonitoring > Metrics with no dimensions N. Virginia

Search for any metric, dimension, resource id or account id

Metric name 1/1 Alarms

TomcatStatus No alarms

Cancel Create widget



**12. If CPU utilization is more than 70%, then it should trigger auto scaling and launch new instance.**

In AWS, **Auto Scaling** automatically adjusts the number of EC2 instances based on demand. To trigger scaling when CPU utilization exceeds 70%, you typically:

1. Create a Launch Template.

Launch template details	Launch template name	Default version	Owner
Launch template ID: lt-0d7708e2150cc8371	my-lt	1	arn:aws:iam::471451201019:root

Launch template version details	Description	Date created	Created by
Version: 1 (Default)	-	2025-10-07T13:52:44.000Z	arn:aws:iam::471451201019:root

Instance details	Storage	Resource tags	Network interfaces	Advanced details
AMI ID: ami-052064a798f08f0d3	Instance type: t2.micro	-	Availability Zone: -	Availability Zone Id: -
Key pair name: amazon_key11	Security groups: -	-	Security group IDs: sg-071d2f2717a53d1f3	-

2. Create an Auto Scaling Group (ASG).

my-asg Capacity overview	Desired capacity	Scaling limits (Min - Max)	Desired capacity type	Status
Desired capacity: 3	Scaling limits (Min - Max): 1 - 3	Desired capacity type: Units (number of instances)	Status: -	-

Date created
Tue Oct 07 2025 19:24:43 GMT+0530 (India Standard Time)

Details	Integrations	Automatic scaling	Instance management	Instance refresh	Activity	Monitoring	Tags - moved
<b>Launch template</b>							

Launch template	AMI ID	Instance type	Owner
Launch template: my-lt	AMI ID: ami-052064a798f08f0d3	Instance type: t2.micro	Owner: arn:aws:iam::471451201019:root

Version	Security groups	Security group IDs	Create time
Default	-	-	Tue Oct 07 2025 19:22:44 GMT+0530 (India Standard Time)

Description	Storage (volumes)	Key pair name	Request Spot Instances
-	-	amazon_key11	No

<b>Network</b>	<b>Subnet ID</b>	<b>Availability Zone distribution</b>	<b>Edit</b>
Availability Zones use1-az1 (us-east-1d) use1-az2 (us-east-1a)	subnet-09bb41f51cb89a526 subnet-0d2d45ea999877b45	Balanced only	
<b>Instance type requirements</b>			
Your Auto Scaling group adheres to the launch template for purchase option and instance type.			
<b>Health checks</b>			
Health check type EC2	Health check grace period 300	<b>Edit</b>	
<b>Instance maintenance policy</b>			
Replacement behavior No policy	Min healthy percentage -	Max healthy percentage -	<b>Edit</b>
<b>Capacity Reservation preference</b>			
Preference Default	Capacity Reservation IDs -	Resource Groups -	<b>Edit</b>

<b>Advanced configurations</b>				<b>Edit</b>
Instance scale-in protection Not protected from scale in	Termination policies Default	Maximum instance lifetime -	Service-linked role arn:aws:iam::471451201019:role/aws-service-role/autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling	
Placement group -	Suspended processes -	Default cooldown 300	Default instance warmup Disabled	
<p><span>ⓘ Tags have been moved to its own tab</span> <span>View tags tab</span></p>				

### 3. Create a CloudWatch alarm for CPU utilization > 70%.

Create ok alarm for cpu utilization <70:

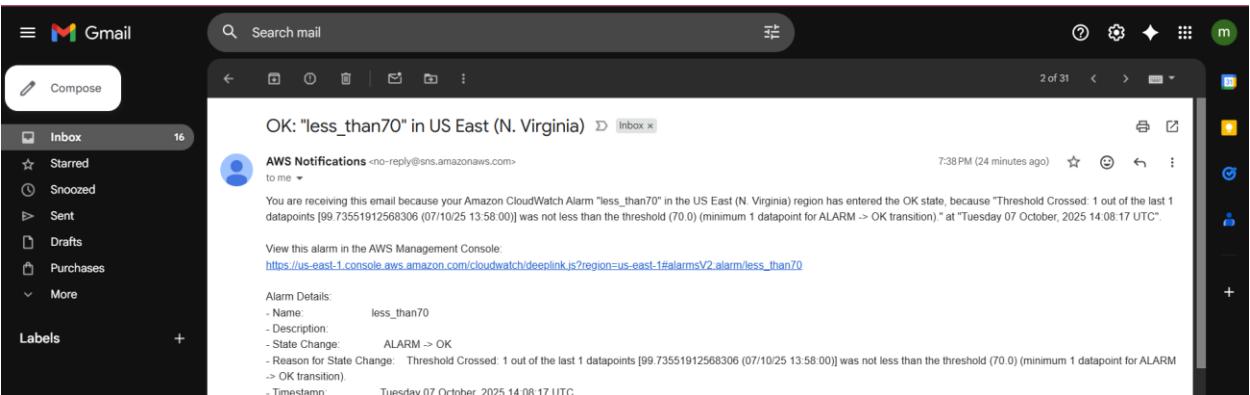
<b>Conditions</b>			
Threshold type			
<input checked="" type="radio"/> Static Use a value as a threshold		<input type="radio"/> Anomaly detection Use a band as a threshold	
<b>Whenever CPUUtilization is...</b> Define the alarm condition.			
<input type="radio"/> Greater > threshold		<input type="radio"/> Greater/Equal >= threshold	
<input type="radio"/> Lower/Equal <= threshold		<input checked="" type="radio"/> Lower < threshold	
<b>than...</b> Define the threshold value. <input type="text" value="70"/> <small>Must be a number.</small>			
<b>► Additional configuration</b>			
<a href="#">Cancel</a> <a href="#">Skip to Preview and update</a> <a href="#">Next</a>			

Create in-Alarm for cpu utilization >70:

Name	State	Last state update (UTC)	Conditions	Actions
less_than70	OK	2025-10-07 14:08:17	CPUUtilization < 70 for 1 datapoints within 5 minutes	Actions enabled
my-alarm-for-cpu-utilization	Insufficient data	2025-10-07 14:29:21	CPUUtilization > 50 for 1 datapoints within 1 minute	Actions enabled

4. Attach the CloudWatch alarm to scaling policies that increase or decrease the number of instances.

Set the notification while creating the alarm by creating the topic and create subscription and provide the endpoint, then we will get the notification whenever alarm triggered:



-----complete-----