

## AWS Task-2

1. The VPC should contain 512 IP addresses.
2. I am using the IP range: 10.5.126.0.
3. CIDR block: 10.5.126.0/23 (provides 512 IP addresses).

### VPC Creation

The screenshot shows the 'Create VPC' wizard in the AWS VPC console. The 'VPC Settings' step is selected. The configuration is as follows:

- Resources to create:** VPC only
- Name tag - optional:** AWS-VPC-1
- IPv4 CIDR block:** 10.5.126.0/23
- IPv6 CIDR block:** No IPv6 CIDR block
- Tenancy:** Default

The browser address bar shows the URL: `us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#CreateVpc:createMode=vpcOnly`. The AWS logo and search bar are visible at the top. The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time: 30-12-2024.

Screenshot of the AWS VPC creation wizard:

**IPv4 CIDR**: 10.5.126.0/23

**IPv6 CIDR block**: No IPv6 CIDR block

**Tenancy**: Default

**Tags**: AWS-VPC-1

**Create VPC**

Screenshot of the AWS VPC dashboard for the newly created VPC:

**VPC ID**: vpc-0d1f7665b71a6f316

**State**: Available

**DNS resolution**: Enabled

**Main network ACL**: acl-0a19783ebfc46b9e7

**IPv6 CIDR**: -

**Block Public Access**: Off

**DHCP option set**: dopt-003a53198bcfc0b07

**IPv4 CIDR**: 10.5.126.0/23

**Route 53 Resolver DNS Firewall rule groups**: -

**DNS hostnames**: Disabled

**Main route table**: rtb-09508c866f83bf1d7

**IPv6 pool**: -

**Owner ID**: 905418201986

**Actions**

## Internet Gateway creation

The screenshot shows the AWS VPC dashboard with the 'Internet gateways' section selected. A table lists one Internet gateway:

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-01907f92c32f10b85	Attached	vpc-07f2dc389716a7f01	905418201986

Below the table, a message says "Select an internet gateway above".

The screenshot shows the "Create internet gateway" wizard. Step 1: Internet gateway settings. It asks for a name tag:

**Internet gateway settings**

**Name tag**  
Creates a tag with a key of 'Name' and a value that you specify.  
AWS-INTERNET-GATEWAY

The screenshot shows the "Create internet gateway" wizard. Step 2: Tags - optional. It shows a tag being added:

**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
Name	AWS-INTERNET-GATEWAY

Add new tag

You can add 49 more tags.

Cancel Create internet gateway

The screenshot shows the AWS navigation bar at the bottom of the browser window, displaying various icons and the current date and time.

## Internet gateway created

The screenshot shows the AWS VPC dashboard. A green notification bar at the top right says: "The following internet gateway was created: igw-03764ac99c4627a44 - AWS-INTERNET-GATEWAY. You can now attach to a VPC to enable the VPC to communicate with the internet." Below this, the Internet gateway details are shown: Internet gateway ID is igw-03764ac99c4627a44, State is Detached, VPC ID is -, and Owner is 905418201986. The Tags section is empty. On the right, there is an "Actions" button with options: "Attach to a VPC", "Actions ▾", "Manage tags", and "Delete". The left sidebar shows navigation links for VPC, Security, and PrivateLink and Lattice.

## Attaching Internet Gateway to the VPC

The screenshot shows the AWS VPC dashboard. A green notification bar at the top right says: "The following internet gateway was created: igw-03764ac99c4627a44 - AWS-INTERNET-GATEWAY. You can now attach to a VPC to enable the VPC to communicate with the internet." Below this, the Internet gateway details are shown: Internet gateway ID is igw-03764ac99c4627a44, State is Detached, VPC ID is -, and Owner is 905418201986. The Tags section is empty. On the right, there is an "Actions" button with options: "Attach to VPC", "Detach from VPC", "Manage tags", and "Delete". A context menu is open over the "Attach to VPC" button, listing "Attach to VPC", "Detach from VPC", "Manage tags", and "Delete". The left sidebar shows navigation links for VPC, Security, and PrivateLink and Lattice.

The screenshot shows the 'Attach to VPC' configuration page for an Internet Gateway (igw-03764ac99c4627a44). The 'Available VPCs' section is visible, showing a search bar and a dropdown menu with one item: 'vpc-0d1f7665b71a6f316 - AWS-VPN-1'. At the bottom right are 'Cancel' and 'Attach internet gateway' buttons.

The screenshot shows the same configuration page, but the 'Available VPCs' dropdown now contains two items: 'vpc-0d1f7665b71a6f316 - AWS-VPN-1' and 'vpc-0d1f7665b71a6f316 - AWS-VPN-2'. The 'Attach internet gateway' button is highlighted in orange.

The screenshot shows the configuration page again, but the 'Available VPCs' dropdown now only contains 'vpc-0d1f7665b71a6f316 - AWS-VPN-2'. The 'Attach internet gateway' button is still highlighted in orange.

## Attached Internet Gateway to the AWS-VPC-1

The screenshot shows the AWS VPC dashboard. On the left, a sidebar lists various VPC components: EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, NAT gateways, Peering connections), Security (Network ACLs, Security groups), and PrivateLink and Lattice. The main content area displays an Internet gateway named "igw-03764ac99c4627a44" successfully attached to a VPC. The "Details" section shows the Internet gateway ID (igw-03764ac99c4627a44), State (Attached), VPC ID (vpc-0d1f7665b71a6f316 | AWS-VPC-1), and Owner (905418201986). The "Tags" section contains a single tag: Name = AWS-INTERNET-GATEWAY. A green banner at the top indicates the attachment was successful. The bottom of the screen shows the Windows taskbar with various pinned icons.

## Creating Subnet1 with 256 IPs

The screenshot shows the AWS VPC dashboard. The sidebar includes EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections), Security (Network ACLs, Security groups), and PrivateLink and Lattice. The main content area shows a table of existing subnets under the heading "Subnets (6) Info". The table columns are: Name, Subnet ID, State, VPC, Block Public..., and IPv4 CIDR. The subnets listed are: subnet-07c77b859be636e7c, subnet-08d6a59dac11aa09b, subnet-086c940b64a97a6d4, subnet-087f1289ec00d8051, subnet-0f67565ca4322f8a2, and subnet-0e07ae45177b4d083, all in an Available state. Below the table, a section titled "Select a subnet" is visible. The bottom of the screen shows the Windows taskbar with various pinned icons.

## Select VPC ID

The screenshot shows the AWS VPC console interface for creating a new subnet. In the top left, under 'VPC ID', the dropdown menu is open, showing 'vpc-0df7665b71a6f316 (AWS-VPC-1)'. On the right, the 'Create subnet' section is visible, containing instructions and optional IPv6 CIDR block fields. The bottom navigation bar includes CloudShell, Feedback, and various AWS service icons.

## Public Subnet1- details

This screenshot continues from the previous one, showing the detailed configuration for the 'Public-Subnet-1'. It includes fields for 'Subnet name' (Public-Subnet-1), 'Availability Zone' (US East (N. Virginia) / us-east-1a), 'IPv4 VPC CIDR block' (10.5.126.0/23), and 'IPv4 subnet CIDR block' (10.5.126.0/24). A 'Tags - optional' section is also present. The right side of the screen displays the same 'Create subnet' information as the first screenshot. The bottom navigation bar is identical to the first one.

## Subnet1 CIDR-10.5.126.0/24 contains-256 ips

The screenshot shows the 'Create subnet' wizard in the AWS VPC console. The 'Availability Zone' is set to 'US East (N. Virginia) / us-east-1a'. The 'IPv4 VPC CIDR block' is '10.5.126.0/23'. The 'IPv4 subnet CIDR block' is '10.5.126.0/24'. A single tag 'Public-Subnet-1' is added under 'Tags - optional'. The 'Create subnet' button is highlighted.

## Created subnet1

The screenshot shows the 'Subnets' page in the AWS VPC dashboard. It displays a success message: 'You have successfully created 1 subnet: subnet-Oedea1299b753342d'. The table lists one subnet: 'Public-Subnet-1' with Subnet ID 'subnet-Oedea1299b753342d', State 'Available', VPC 'vpc-0d1f7665b71a6f316 | AWS...', Block Public 'Off', and IPv4 CIDR '10.5.126.0/24'. The 'Actions' and 'Create subnet' buttons are visible at the top right.

## Creating Subnet-2 with 256 IPs

### Select VPC ID

The screenshot shows the 'Create subnet' page in the AWS VPC console. At the top, there is a search bar and a breadcrumb navigation path: 'VPC > Subnets > Create subnet'. Below this, the title 'Create subnet' is displayed with a 'Info' link. A section titled 'VPC' contains the heading 'VPC ID' and the sub-instruction 'Create subnets in this VPC.' A dropdown menu labeled 'Select a VPC' is open, showing two options: 'vpc-07f2dc389716a7f01' (172.31.0.0/16) and 'vpc-0d1f7665b71a6f316 (AWS-VPC-1)' (10.5.126.0/23). A note at the bottom of this section says 'Select a VPC first to create new subnets.' Below the dropdown is a button 'Add new subnet'. At the bottom right of the page are 'Cancel' and 'Create subnet' buttons.

### Private Subnet-2 details

The screenshot shows the 'Create subnet' page with the 'Subnet settings' section expanded. The title 'Subnet settings' is at the top, followed by the instruction 'Specify the CIDR blocks and Availability Zone for the subnet.' Below this, the 'Subnet 1 of 1' section is shown. It includes a 'Subnet name' field containing 'Private-Subnet-2', a note that names can be up to 256 characters long, and an 'Availability Zone' dropdown set to 'US East (N. Virginia) / us-east-1a'. Under 'IPv4 VPC CIDR block', it says 'Info' and 'Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.', with a dropdown showing '10.5.126.0/23'. The 'IPv4 subnet CIDR block' section shows '10.5.127.0/24' with a note '256 IPs'. At the bottom, there is a 'Tags - optional' section with a key 'Name' and value 'Private-Subnet-2'. The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time '30-12-2024'.

## Subnet 2 CIDR-10.5.127.0/24 contains-256 ips

The screenshot shows the 'Create subnet' wizard in the AWS VPC console. The steps completed are:

- Availability Zone**: US East (N. Virginia) / us-east-1a
- IPv4 VPC CIDR block**: 10.5.126.0/23
- IPv4 subnet CIDR block**: 10.5.127.0/24 (256 IPs)
- Tags - optional**: Private-Subnet-2

At the bottom right, there are 'Cancel' and 'Create subnet' buttons.

## Created Private subnet-2

The screenshot shows the 'Subnets' page in the AWS VPC dashboard. A success message indicates "You have successfully created 1 subnet: subnet-0d0f49f0802297f61". The table displays the following subnet information:

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
Private-Subnet-2	subnet-0d0f49f0802297f61	Available	vpc-0d1f7665b71a6f316   AWS...	Off	10.5.127.0/24

At the bottom right, there are 'Actions' and 'Create subnet' buttons.

You have successfully created 1 subnet: subnet-0d0f49f0802297f61

### Subnets (2/8) Info

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-086c940b64a97a6d4	Available	vpc-07f2dc389716a7f01	Off	172.31.0.0/16
-	subnet-087f1289ec00d8051	Available	vpc-07f2dc389716a7f01	Off	172.31.64.0/20
-	subnet-0f67565ca4322f8a2	Available	vpc-07f2dc389716a7f01	Off	172.31.16.0/20
-	subnet-0e07ae45177b4d083	Available	vpc-07f2dc389716a7f01	Off	172.31.80.0/20
<input checked="" type="checkbox"/> Public-Subnet-1	subnet-0dea1299b753342d	Available	vpc-0d1f7665b71a6f316   AWS...	Off	10.5.126.0/20
<input checked="" type="checkbox"/> Private-Subnet-2	subnet-0d0f49f0802297f61	Available	vpc-0d1f7665b71a6f316   AWS...	Off	10.5.127.0/20

Subnets: subnet-0dea1299b753342d, subnet-0d0f49f0802297f61

## AWS-VPC-1 Has 2 subnets

You have successfully created 1 subnet: subnet-0d0f49f0802297f61

### Subnets (2) Info

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0dea1299b753342d	Available	vpc-0d1f7665b71a6f316   AWS...	Off	10.5.126.0/20
-	subnet-0d0f49f0802297f61	Available	vpc-0d1f7665b71a6f316   AWS...	Off	10.5.127.0/20

Select a subnet

## Create Route Table

VPC dashboard < Actions ▾ Create route table

EC2 Global View Filter by VPC

vpc-0d1f7665b71a6f316  
AWS-VPC-1  
Owner: 905418201986

Virtual private cloud ▼

Your VPCs  
Subnets  
**Route tables**  
Internet gateways  
Egress-only internet gateways  
Carrier gateways  
DHCP option sets  
Elastic IPs  
Managed prefix lists  
NAT gateways  
Peering connections

Security ▼

CloudShell Feedback

21°C Haze

Search [Alt+S]

Last updated 37 minutes ago

No matching resource found

Select a route table

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ENG IN 23:47 30-12-2024

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#CreateRouteTable:

AWS > 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  Remove

Add new tag

You can add 49 more tags.

Cancel Create route table

CloudShell Feedback

21°C Haze

Search [Alt+S]

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ENG IN 23:53 30-12-2024

## Creating Route Table-1 for public subnet-1

The screenshot shows the AWS VPC console interface. On the left, a navigation sidebar includes sections for VPC dashboard, EC2 Global View, Virtual private cloud (with sub-options like Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, and Peering connections), Security (Network ACLs, Security groups), and PrivateLink and Lattice. The main content area displays a success message: "Route table rtb-0c9719deb1d774290 | Route-table-1 was created successfully." Below this, the "rtb-0c9719deb1d774290 / Route-table-1" page is shown. The "Details" section shows the Route table ID (rtb-0c9719deb1d774290), Main status (No), and Owner ID (905418201986). The "Routes" tab is selected, showing one route entry: Destination 10.5.126.0/23, Target local, Status Active, and Propagated No. The bottom of the screen shows the Windows taskbar with various pinned icons.

## Connecting The Route table to the internet gateway by edit routes

This screenshot is identical to the previous one, showing the creation of Route Table-1. However, it also includes a step where the route table is being connected to an internet gateway. The "Edit routes" button in the "Routes" section is highlighted, indicating the next action. The rest of the interface and taskbar are the same as the first screenshot.

The screenshot shows the AWS VPC console with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-0c9719deb1d774290>. The page title is "Edit routes". The main content area displays a table with one row:

Destination	Target	Status	Propagated
10.5.126.0/23	local	Active	No

Below the table are "Add route" and "Save changes" buttons. The browser's address bar shows the same URL. The top navigation bar includes tabs like "VPC", "Route tables", and "Edit routes". The status bar at the bottom right shows "N. Virginia" and the user "srikanth kolluri".

The screenshot shows the AWS CloudShell interface with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-0c9719deb1d774290>. The top bar includes "CloudShell" and "Feedback" buttons. The browser's address bar shows the same URL. The status bar at the bottom right shows "ENG IN" and the date "31-12-2024".

## Add route → Select Internet Gateway

The screenshot shows the AWS VPC console with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-0c9719deb1d774290>. The page title is "Edit routes". The main content area displays a table with one row:

Destination	Target	Status	Propagated
10.5.126.0/23	local	Active	No

Below the table is an "Add route" button. To the right of the table are "Cancel", "Preview", and "Save changes" buttons. A dropdown menu is open under the "Target" column, showing the following options:

- Carrier Gateway
- Core Network
- Egress Only Internet Gateway
- Gateway Load Balancer Endpoint
- Instance
- Internet Gateway
- local
- NAT Gateway
- Network Interface
- Outpost Local Gateway
- Peering Connection
- Transit Gateway

The browser's address bar shows the same URL. The top navigation bar includes tabs like "VPC", "Route tables", and "Edit routes". The status bar at the bottom right shows "N. Virginia" and the user "srikanth kolluri".

The screenshot shows the AWS CloudShell interface with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-0c9719deb1d774290>. The top bar includes "CloudShell" and "Feedback" buttons. The browser's address bar shows the same URL. The status bar at the bottom right shows "ENG IN" and the date "31-12-2024".

The screenshot shows the AWS VPC Edit Routes interface. A route is being added for the destination 10.5.126.0/23. The target is set to 'local' (selected from a dropdown) and is marked as 'Active'. The status is 'Propagated' and 'No' is selected for 'Propagate to BGP'. The target is then changed to 'Internet Gateway' (selected from a dropdown) and is marked as 'In Progress'. The status is 'Propagated' and 'No' is selected for 'Propagate to BGP'. A search bar contains 'igw-' and the result 'igw-03764ac99c4627a44 (AWS-INTERNET-GATEWAY)' is shown. At the bottom, there are 'Add route', 'Cancel', 'Preview', and 'Save changes' buttons.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences  
1 NZ - SL Game score ENG IN 31-12-2024

## Save Changes

The screenshot shows the AWS VPC console's 'Edit routes' page for a specific route table. The top navigation bar includes tabs for 'aws' and 'Search'. The main content area displays two routes:

Destination	Target	Status	Propagated
10.5.126.0/23	local	Active	No
0.0.0.0/0	Internet Gateway	-	No

Below the table, there are search input fields for both destination and target, and a 'Remove' button for the second route. At the bottom, there are 'Add route', 'Cancel', 'Preview', and 'Save changes' buttons.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 31-12-2024 0002

## Allowed Internet Access to the Route Table-1

The screenshot shows the AWS VPC console interface. The left sidebar is collapsed. The main content area displays the details for Route Table-1, which was updated successfully. The table includes fields for Route table ID (rtb-0c9719deb1d774290), Main status (No), Owner ID (905418201986), Explicit subnet associations (empty), and Edge associations (empty). Below this, the 'Routes' tab is selected, showing two routes: one to the internet gateway (igw-03764ac99c4627a44) and one to the local subnet.

## Creating Route table-2 for Private Subnet-2

The screenshot shows the 'Create route table' wizard. In the 'Route table settings' step, a name 'Route-table-2' is entered for the route table. The VPC dropdown is set to 'vpc-0d1f7665b71a6f316 (AWS-VPC-1)'. In the 'Tags' step, a single tag 'Name: Route-table-2' is added. The final step shows the 'Create route table' button highlighted.

The screenshot shows the AWS VPC console with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#RouteTableDetails:RouteTableId=rtb-023c4b98bf5bcd41>. The page displays a success message: "Route table rtb-023c4b98bf5bcd41 | Route-table-2 was created successfully." Below this, the details for "rtb-023c4b98bf5bcd41 / Route-table-2" are shown, including its ID, Main status, Owner ID, and no explicit subnet associations or edge associations. The "Routes" tab is selected, showing one route entry: Destination 10.5.126.0/23, Target local, Status Active, and Propagated No. The left sidebar shows the navigation menu for VPC, including Virtual private cloud, Security, and PrivateLink and Lattice.

## Subnet Associations

### Associating Route-table-1 with the public subnet public-subnet1

The screenshot shows the AWS VPC console with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#RouteTables:vpcId=vpc-0d1f7665b71a6f316>. The "Route tables (1/3)" table lists three route tables: "rtb-09508c868f83bfd17" (Main Yes, VPC vpc-0d1f7665b71a6f316), "Route-table-1" (selected, Main No, VPC vpc-0d1f7665b71a6f316), and "rtb-023c4b98bf5bcd41" (Main No, VPC vpc-0d1f7665b71a6f316). The "rtb-0c9719deb1d774290 / Route-table-1" details page is shown, with the "Subnet associations" tab selected. It shows no explicit subnet associations or edge associations. The left sidebar shows the navigation menu for VPC, including Virtual private cloud, Security, and PrivateLink and Lattice.

Select Route table1→click on subnet associations→Edit subnet associations→select public subnet1

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

Last updated 15 minutes ago Actions Create route table

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC
-	rtb-09508c868f83bfd17	-	-	Yes	vpc-0d1f7665b71a6f316   A'
<input checked="" type="checkbox"/> Route-table-1	rtb-0c9719deb1d774290	-	-	No	vpc-0d1f7665b71a6f316   A'
Route-table-2	rtb-023c4b98bf5bcd41	-	-	No	vpc-0d1f7665b71a6f316   A'

**Subnets without explicit associations (2)**

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Public-Subnet-1	subnet-0dea1299b753342d	10.5.126.0/24	-
Private-Subnet-2	subnet-0d0f49f0802297f61	10.5.127.0/24	-

**Edit subnet associations**

Change which subnets are associated with this route table.

**Available subnets (1/2)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/> Public-Subnet-1	subnet-0dea1299b753342d	10.5.126.0/24	-	Main (rtb-09508c868f83bfd17)
<input type="checkbox"/> Private-Subnet-2	subnet-0d0f49f0802297f61	10.5.127.0/24	-	Main (rtb-09508c868f83bfd17)

**Selected subnets**

subnet-0dea1299b753342d / Public-Subnet-1

Cancel **Save associations**

You have successfully updated subnet associations for rtb-0c9719deb1d774290 / Route-table-1.

Route tables (1/3) [Info](#)

Last updated less than a minute ago [Actions](#) [Create route table](#)

VPC	Name	Route table ID	Explicit subnet assoc...	Main	VPC
vpc-0d1f7665b71a6f316	-	rtb-09508c868fb83bfd17	-	Yes	vpc-0d1f7665b71a6f316   A'
	Route-table-1	rtb-0c9719deb1d774290	subnet-Odea1299b753342d / Public-Subnet-1	No	vpc-0d1f7665b71a6f316   A'
	Route-table-2	rtb-023c4b98bf5bcd41	-	No	vpc-0d1f7665b71a6f316   A'

rtb-0c9719deb1d774290 / Route-table-1

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

**Details**

Route table ID rtb-0c9719deb1d774290	Main No	Explicit subnet associations subnet-Odea1299b753342d / Public-Subnet-1	Edge associations -
VPC vpc-0d1f7665b71a6f316	Owner ID AWS Lambda	© 2024, Amazon Web Services, Inc. or its affiliates.	

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 00:23 31-12-2024

## Associating Route-table-2 with the private subnet private-subnet-2

You do not have any subnet associations.

Route tables (1/3) [Info](#)

Last updated 1 minute ago [Actions](#) [Create route table](#)

VPC	Name	Route table ID	Explicit subnet assoc...	Main	VPC
vpc-0d1f7665b71a6f316	-	rtb-09508c868fb83bfd17	-	Yes	vpc-0d1f7665b71a6f316   A'
	Route-table-1	rtb-0c9719deb1d774290	subnet-Odea1299b753342d / Public-Subnet-1	No	vpc-0d1f7665b71a6f316   A'
	Route-table-2	rtb-023c4b98bf5bcd41	-	No	vpc-0d1f7665b71a6f316   A'

**Subnets without explicit associations (1)**

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Private-Subnet-2	subnet-0d0f49f0802297f61	10.5.127.0/24	-

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 00:25 31-12-2024

**Select Route table-2→click on subnet associations→Edit subnet associations→select private subnet-2**

The screenshot shows the 'Edit subnet associations' dialog in the AWS VPC console. The table lists two subnets:

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
Public-Subnet-1	subnet-0dea1299b753342d	10.5.126.0/24	-	rtb-0c9719deb1d774290 / Route-table-2
<b>Private-Subnet-2</b>	subnet-0d0f49f0802297f61	10.5.127.0/24	-	Main (rtb-09508c868f83bfd17)

The 'Selected subnets' section contains the selected subnet: **subnet-0d0f49f0802297f61 / Private-Subnet-2**.

Buttons at the bottom: **Cancel** and **Save associations**.

The screenshot shows the VPC dashboard. A success message says: **You have successfully updated subnet associations for rtb-023c4b98bf5bcd41 / Route-table-2.**

The 'Route tables' table shows three route tables:

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
-	rtb-09508c868f83bfd17	-	-	Yes	vpc-0d1f7665b71a6f316   A'
Route-table-1	rtb-0c9719deb1d774290	subnet-0dea1299b753342d	-	No	vpc-0d1f7665b71a6f316   A'
<b>Route-table-2</b>	<b>rtb-023c4b98bf5bcd41</b>	<b>subnet-0d0f49f0802297f61</b>	-	No	<b>vpc-0d1f7665b71a6f316   A'</b>

The details for route table **rtb-023c4b98bf5bcd41 / Route-table-2** show:

- Details**: Route table ID: rtb-023c4b98bf5bcd41, Main: No, Owner ID: vpc-0d1f7665b71a6f316.
- Explicit subnet associations**: subnet-0d0f49f0802297f61 / Private-Subnet-2.
- Edge associations**: -

## Launching EC2 Instance with VPC and Public subnet-1

The screenshot shows the AWS Lambda console with the 'Launch an instance' wizard open. The top navigation bar includes tabs for 'EC2' and 'Instances'. The main area is titled 'Launch an instance' with a sub-section 'Name and tags'. A text input field contains 'AWS-Linux-EC2'. To the right is a button 'Add additional tags'. Below this is a section titled 'Application and OS Images (Amazon Machine Image)'. It features a search bar and a grid of icons for various AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian. A callout box highlights the 'Free tier' information: 'In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month. TGA hours of regular t2.micro usage are charged at \$0.016/hour'. On the right side, there's a summary panel with details like 'Number of instances: 1', 'Software Image (AMI): Amazon Linux 2023 AMI 2023.6.2...', 'Virtual server type (instance type): t2.micro', 'Firewall (security group): New security group', 'Storage (volumes): 1 volume(s) - 8 GiB', and a large orange 'Launch instance' button.

This screenshot continues the 'Launch an instance' wizard. In the 'Amazon Machine Image (AMI)' section, it shows the 'Amazon Linux 2023 AMI' selected. The details listed are: 'ami-01816d07b1128cd2d (64-bit (x86), uefi-preferred) / ami-02dcfe5d1d39baa4e (64-bit (Arm), uefi)', 'Virtualization: hvm', 'ENA enabled: true', and 'Root device type: ebs'. The 'Free tier eligible' status is indicated. Below this, the 'Description' section notes that Amazon Linux 2023 is a modern, general purpose Linux-based OS. The 'Architecture' dropdown is set to '64-bit (x86)'. The 'Boot mode' is 'uefi-preferred'. The 'AMI ID' is 'ami-01816d07b1128cd2d'. The 'Username' is 'ec2-user' with a 'Verified provider' badge. The 'Summary' panel on the right remains the same as in the previous screenshot, showing the launch configuration and the 'Launch instance' button.

**Additional costs apply for AMIs with pre-installed software**

**Key pair (login)** Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

**Key pair name - required**

mounikakeypair

**Network settings** Info

**Network** Info  
vpc-07f2dc589716a7f01

**Subnet** Info  
No preference (Default subnet in any availability zone)

**Auto-assign public IP** Info  
Enable  
Additional charges apply when outside of free tier allowance

**Firewall (security groups)** Info  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group  Select existing security group

**Summary**

Number of instances Info  
1

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.6.2... read more  
ami-01816d07b11280cd2d

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 00:39 31-12-2024

## Edit Network Settings → Select VPC → Select Public-Subnet-1

**Network settings** Info

**VPC - required** Info  
vpc-0d1f7665b71a6f316 (AWS-VPC-1)  
10.5.126.0/23

vpc-0d1f7665b71a6f316 (AWS-VPC-1)  
10.5.126.0/23

vpc-07f2dc589716a7f01 (default)

**Enable**  
Additional charges apply when outside of free tier allowance

**Firewall (security groups)** Info  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group  Select existing security group

**Security group name - required**  
launch-wizard-11

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_-/!@#%^&{}\$\*

**Description - required** Info  
launch-wizard-11 created 2024-12-30T19:05:40.812Z

**Summary**

Number of instances Info  
1

t2.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 00:44 31-12-2024

## Enabled Auto Assign public ip

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Network settings' section, 'Auto-assign public IP' is set to 'Enable'. A tooltip for the 'Free tier' explains that it includes 750 hours of t2.micro usage per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

**Network settings**

VPC - required: vpc-0d1f7665b71a6f316 (AWS-VPC-1) 10.5.126.0/23

Subnet: subnet-0dea1299b753342d Public-Subnet-1 VPC: vpc-0d1f7665b71a6f316 Owner: 905418201986 Availability Zone: us-east-1a Zone type: Availability Zone IP addresses available: 251 CIDR: 10.5.126.0/24

Create new subnet

Auto-assign public IP: Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups): Create security group

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Security group name - required: launch-wizard-11

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_~!@#\$%^&\_!\$\*

Description - required: launch-wizard-11 created 2024-12-30T19:05:40.812Z

**Summary**

Number of instances: 1

t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro) in the Regions in which t2.micro is unavailable instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Launch instance

Preview code

## Instance Launched

The screenshot shows the AWS EC2 'Launch an instance' wizard with a green success message: 'Successfully initiated launch of instance i-0d01292cb15dc4d5d'. Below the message, there's a 'Launch log' link. The 'Next Steps' section contains several cards:

- Create billing and free tier usage alerts**: To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. Link: [Create billing alerts](#).
- Connect to your instance**: Once your instance is running, log into it from your local computer. Links: [Connect to instance](#), [Learn more](#).
- Connect an RDS database**: Configure the connection between an EC2 instance and a database to allow traffic flow between them. Links: [Connect an RDS database](#), [Create a new RDS database](#), [Learn more](#).
- Create EBS snapshot policy**: Create a policy that automates the creation, retention, and deletion of EBS snapshots. Link: [Create EBS snapshot policy](#).

Below these cards are links for 'Manage detailed monitoring', 'Create Load Balancer', 'Create AWS budget', and 'Manage CloudWatch alarms'.

**Success**  
Successfully initiated launch of instance i-0d01292cb15dc4d5d

Launch log

**Next Steps**

What would you like to do next with this instance, for example "create alarm" or "create backup"

1 2 3 4 5 6 >

Create billing and free tier usage alerts

Connect to your instance

Connect an RDS database

Create EBS snapshot policy

Manage detailed monitoring

Create Load Balancer

Create AWS budget

Manage CloudWatch alarms

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with sections like Dashboard, EC2 Global View, Events, Instances (with sub-options like Instances, Instance Types, Launch Templates, etc.), Images, AMIs, AMI Catalog, Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security. The main content area is titled 'Instances (1/1) Info'. It shows a table with one row for 'AWS-Linux-EC2'. The table columns include Instance ID, Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IP. The instance details are expanded, showing the following information:

Details	Value
Instance ID	i-0d01292cb15dc4d5d
Public IPv4 address	98.81.190.36   open address
Private IPv4 addresses	10.5.126.243
IPv6 address	-
Instance state	Running
Public IPv4 DNS	-
Hostname type	ip-10-5-126-243.ec2.internal
Private IP DNS name (IPv4 only)	ip-10-5-126-243.ec2.internal
Instance type	t2.micro
VPC ID	vpc-0d1f7665b71a6f316 (AWS-VPC-1)
Subnet ID	subnet-0deea1299b6753342d (Public-Subnet-1)
Instance ARN	arn:aws:ec2:us-east-1:905418201986:instance/i-0d01292cb15dc4d5d
IAM Role	-
IMDSv2	Required
Operator	-

## The instance is running under the VPC AW-VPC-1, in the public subnet public-subnet-1

This screenshot shows the 'Instance summary for i-0d01292cb15dc4d5d (AWS-Linux-EC2)' page. The left sidebar is identical to the previous screenshot. The main content area is titled 'Instance summary for i-0d01292cb15dc4d5d (AWS-Linux-EC2) info'. It provides a more detailed view of the instance's configuration and network settings. Key details include:

Detail	Value
Instance ID	i-0d01292cb15dc4d5d
Public IPv4 address	98.81.190.36   open address
Private IPv4 addresses	10.5.126.243
IPv6 address	-
Instance state	Running
Public IPv4 DNS	-
Hostname type	ip-10-5-126-243.ec2.internal
Private IP DNS name (IPv4 only)	ip-10-5-126-243.ec2.internal
Instance type	t2.micro
VPC ID	vpc-0d1f7665b71a6f316 (AWS-VPC-1)
Subnet ID	subnet-0deea1299b6753342d (Public-Subnet-1)
Instance ARN	arn:aws:ec2:us-east-1:905418201986:instance/i-0d01292cb15dc4d5d
IAM Role	-
IMDSv2	Required
Operator	-

## Auto assigned Public IP is 98.81.190.36

The screenshot shows the AWS CloudWatch Metrics interface. A single metric named 'CPU Utilization' is displayed across three dimensions: Region (us-east-1), Instance ID (i-0d01292cb15dc4d5d), and Metric Name (CPUUtilization). The chart shows a constant value of 0.0% CPU utilization over a period of 1 hour.

## Connected the EC2 instance

```
ec2-user@ip-10-5-126-243:~
```

Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\mouni> cd Downloads
PS C:\Users\mouni\Downloads> ssh -i "mounikakeypair.pem" ec2-user@98.81.190.36
The authenticity of host '98.81.190.36 (98.81.190.36)' can't be established.
ED25519 key fingerprint is SHA256:d/xuKDMbylw1VN5NQW/IzIoFNuvPHF2zzTDu1ZQrtg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '98.81.190.36' (ED25519) to the list of known hosts.

, #_
 ~\ ####_ Amazon Linux 2023
 ~~ \#####\
 ~~ \|##|
 ~~ \#/ __ https://aws.amazon.com/linux/amazon-linux-2023
 ~~ V~' '-'>
 ~~ / \
 ~~ /_ \
 ~~ /m' \
[ec2-user@ip-10-5-126-243 ~]$
```