

Case Study – 1

PRADEEPPA K

Email: kpradeepachary@gmail.com

Problem Statement:

You work for XYZ Corporation and based on the expansion requirements of your corporation you have been asked to create and set up a distinct Amazon VPC for the production and development team. You are expected to perform the following tasks for the respective VPCs.

Production Network:

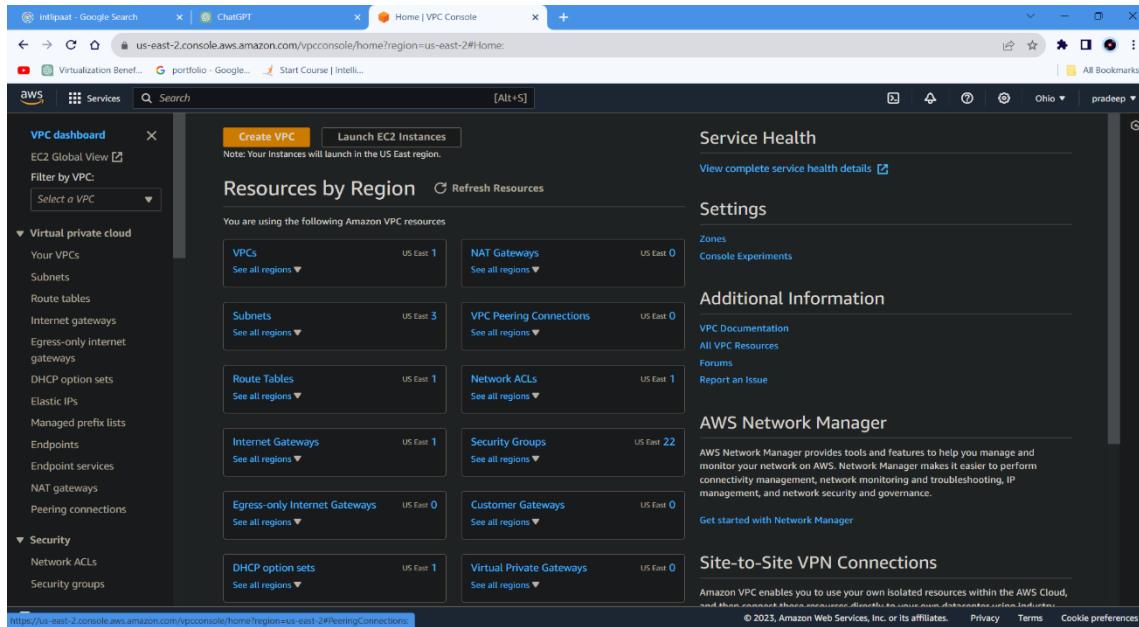
1. Design and build a 4-tier architecture.
2. Create 5 subnets out of which 4 should be private named app1, app2, dbcache and db and one should be public, named web.
3. Launch instances in all subnets and name them as per the subnet that they have been launched in.
4. Allow dbcache instance and app1 subnet to send internet requests.
5. Manage security groups and NACLs.

Development Network:

1. Design and build 2-tier architecture with two subnets named web and db and launch instances in both subnets and name them as per the subnet names.
2. Make sure only the web subnet can send internet requests.
3. Create peering connection between production network and development network.
4. Setup connection between db subnets of both production network and development network respectively.

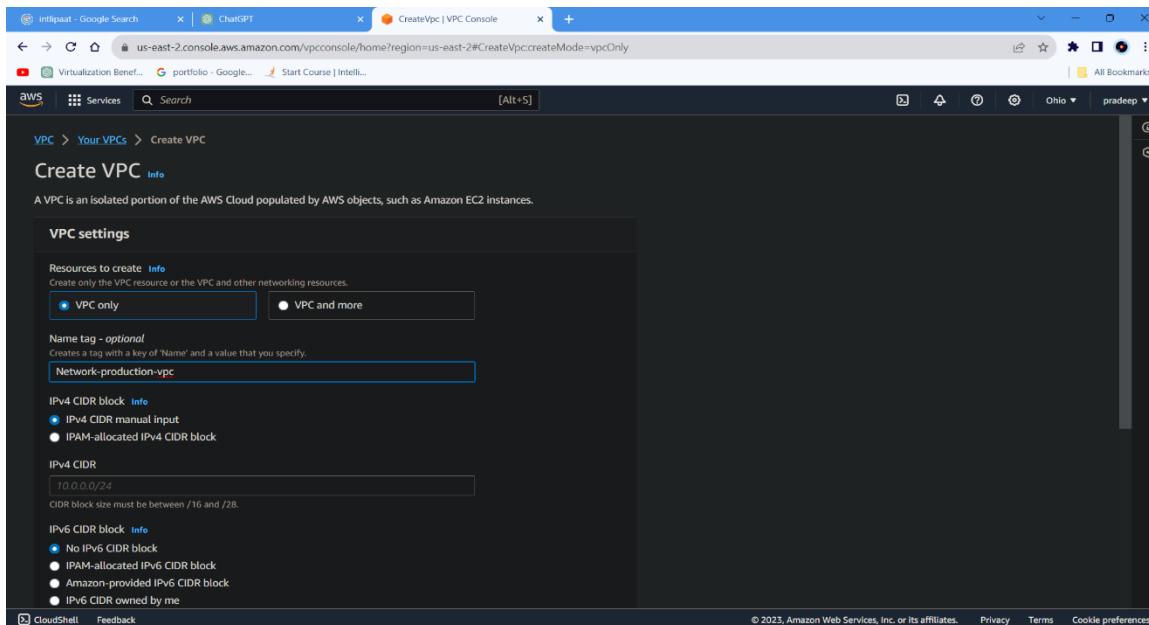
1. First, creation of a Virtual Private Cloud (VPC) is essential for both the production network and development network.

To establish a Virtual Private Cloud (VPC), navigate to the "Create VPC" option and provide the required information.



The screenshot shows the AWS VPC Console Home page. On the left, there's a sidebar with sections like 'Virtual private cloud' (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections, Security, Network ACLs, Security groups) and 'AWS dashboard' (Services, Search). The main area has a 'Create VPC' button and a 'Launch EC2 Instances' button. Below these are sections for 'Resources by Region' (VPCs, Subnets, Route Tables, Internet Gateways, Egress-only Internet Gateways, DHCP option sets, NAT Gateways, NAT Peering Connections, VPC Peering Connections, Network ACLs, Security Groups, Customer Gateways, Virtual Private Gateways) and 'Service Health' (View complete service health details). On the right, there are 'Settings' (Zones, Console Experiments), 'Additional Information' (VPC Documentation, All VPC Resources, Forums, Report an Issue), and 'AWS Network Manager' (Get started with Network Manager). At the bottom, there's a 'Site-to-Site VPN Connections' section and a footer with links to Privacy, Terms, and Cookie preferences.

2. In the process of establishing a VPC for the production network, the VPC is designated as "Network Production VPC," and the corresponding CIDR range is specified.



The screenshot shows the 'Create VPC' wizard, Step 1: VPC settings. It starts with a header 'CreateVpc | VPC Console'. Below it, there's a breadcrumb 'VPC > Your.VPCs > Create VPC'. A note says 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.' The 'VPC settings' section has a 'Resources to create' dropdown set to 'VPC only'. There's a 'Name tag - optional' field with 'Network-production-vpc' entered. Under 'IPv4 CIDR', a field shows '10.0.0.0/24'. Under 'IPv6 CIDR', options include 'No IPv6 CIDR block', 'IPAM-allocated IPv6 CIDR block', 'Amazon-provided IPv6 CIDR block', and 'IPv6 CIDR owned by me'. At the bottom, there are 'CloudShell' and 'Feedback' buttons, and a footer with links to Privacy, Terms, and Cookie preferences.

3. The VPC for the production network has been successfully created.

The screenshot shows the AWS VPC console interface. A green banner at the top indicates "You successfully created vpc-046c5e0929b626cc4 / Network-production-vpc". The main page displays details for the VPC, including its ID (vpc-046c5e0929b626cc4), state (Available), and CIDR range (10.0.0.0/16). It also lists network address usage metrics, Route 53 Resolver DNS Firewall rule groups, and other configuration details like DNS resolution and Main network ACL.

4. And the CIDR range for the VPC has been assigned(10.0.0.0/16).

This screenshot shows the "VpcDetails | VPC Console" page. It highlights the "IPv4 CIDR" section, which is set to 10.0.0.0/16. Other visible details include the VPC ID (dopt-0f99363fb09439d8b), the DHCP option set (rtb-072954dc95ffa65cc), and the Route 53 Resolver DNS Firewall rule groups (disabled).

5. Productions Network VPC.

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with options like 'VPC dashboard', 'EC2 Global View', 'Filter by VPC' (with a dropdown menu), 'Virtual private cloud' (expanded), 'Your VPCs' (selected), 'Subnets', 'Route tables', 'Internet gateways', 'Egress-only internet gateways', 'DHCP option sets', 'Elastic IPs', 'Managed prefix lists', 'Endpoints', 'Endpoint services', 'NAT gateways', 'Peering connections', 'Security' (expanded), 'Network ACLs', and 'Security groups'. The main area is titled 'Your VPCs (2)' and contains a table with two rows:

	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP o
<input type="checkbox"/>	-	vpc-0f4523b2ab35df0f8	Available	172.31.0.0/16	-	dopt-Of
<input type="checkbox"/>	Network-production-vpc	vpc-046c5e0929b626cc4	Available	10.0.0.0/16	-	dopt-Of

Below the table, it says 'Select a VPC above'.

6. In the process of establishing a VPC for the development network, the VPC is designated as "Development Production-VPC," and the corresponding CIDR range is specified as 10.1.0.0/16.

The screenshot shows the 'CreateVpc' VPC Console page. At the top, it says 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.' Below that is a section titled 'VPC settings' with the following fields:

- 'Resources to create' dropdown: 'VPC only' (selected) and 'VPC and more'.
- 'Name tag - optional': 'development-network-vpc'.
- 'IPv4 CIDR block' dropdown: 'IPv4 CIDR manual input' (selected) and 'IPAM-allocated IPv4 CIDR block'. Below it, the 'IPv4 CIDR' field is set to '10.1.0.0/16'.
- 'IPv6 CIDR block' dropdown: 'No IPv6 CIDR block' (selected) and three other options: 'IPAM-allocated IPv6 CIDR block', 'Amazon-provided IPv6 CIDR block', and 'IPv6 CIDR owned by me'.
- 'Tenancy' dropdown: 'Default'.

At the bottom, there are 'CloudShell' and 'Feedback' buttons, and a copyright notice: '© 2023, Amazon Web Services, Inc. or its affiliates.'

7. The VPC for the Development network has been successfully created.

The screenshot shows the AWS VPC Console interface. A green success message at the top center states: "You successfully created vpc-0a6057a365f2533c8 / development-network-vpc". The main content area displays the details of the newly created VPC, "vpc-0a6057a365f2533c8 / development-network-vpc". The "Details" tab is selected, showing the following configuration:

VPC ID	State	DNS hostnames	DNS resolution
vpc-0a6057a365f2533c8	Available	Disabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	dopt-0f99365fb09439d8b	rtb-09d33f5523e268461	acl-074952e8b75e0ad61
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR
No	10.1.0.0/16	-	-
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	
Disabled	-	626130759947	

Below the details, there are tabs for "Resource map", "CIDRs", "Flow logs", "Tags", and "Integrations". The "Resource map" tab is selected, showing four cards: "VPC", "Subnets (0)", "Route tables (1)", and "Networks (0)". The "VPC" card shows "Your AWS virtual network". The "Subnets (0)" card shows "Subnets within this VPC". The "Route tables (1)" card shows "Route network traffic to resources". The "Networks (0)" card shows "Connect to your VPC".

8.

9.

VPC ID: vpc-046c5e0929b626cc4 (Network-production-vpc)

Associated VPC CIDRs: 10.0.0.0/16

Subnet settings

Subnet 1 of 1

Subnet name: my-subnet-01

Availability Zone: No preference

IPv4 VPC CIDR block: 10.0.0.0/16

IPv4 subnet CIDR block: 10.0.0.0/20

10.

Subnet settings

Subnet 1 of 1

Subnet name: app1

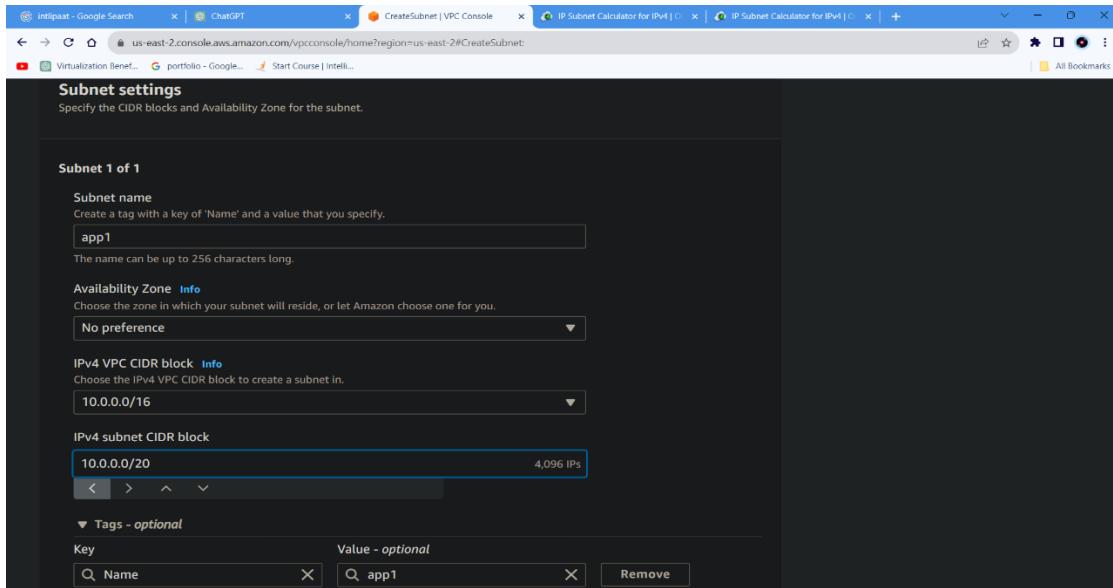
Availability Zone: No preference

IPv4 VPC CIDR block: 10.0.0.0/16

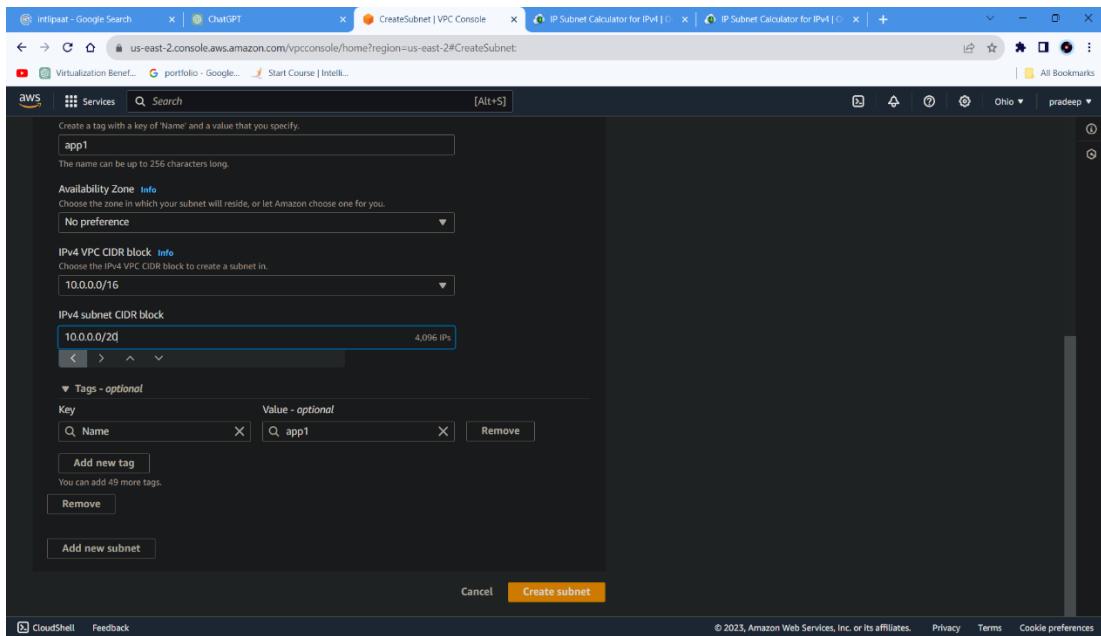
IPv4 subnet CIDR block: 10.0.0.0/20

Tags - optional:

Key	Value - optional
Name	app1



11.



12.

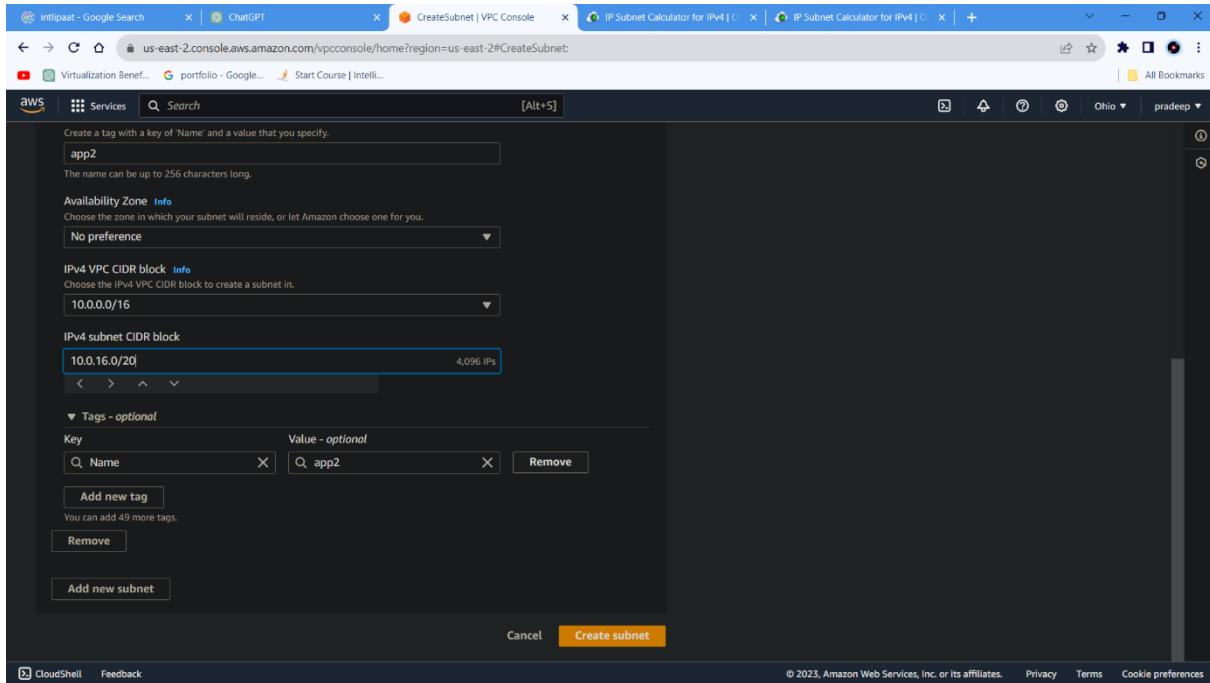
The screenshot shows the AWS VPC Subnets page. A green banner at the top indicates "You have successfully created 1 subnet: subnet-0b13aa0d4eb8c74a9". The main table lists four subnets, including the newly created one named "app1".

Name	Subnet ID	State	VPC	IPv4 CIDR
-	subnet-07684ac46adbca72	Available	vpc-0f4523b2ab35df0f8	172.31.32.0/20
-	subnet-0406e88e14a480286	Available	vpc-0f4523b2ab35df0f8	172.31.16.0/20
-	subnet-0631dfa5d697ee5	Available	vpc-0f4523b2ab35df0f8	172.31.0.0/20
app1	subnet-0b13aa0d4eb8c74a9	Available	vpc-0f46c5e0929b626cc4 Net...	10.0.0.0/20

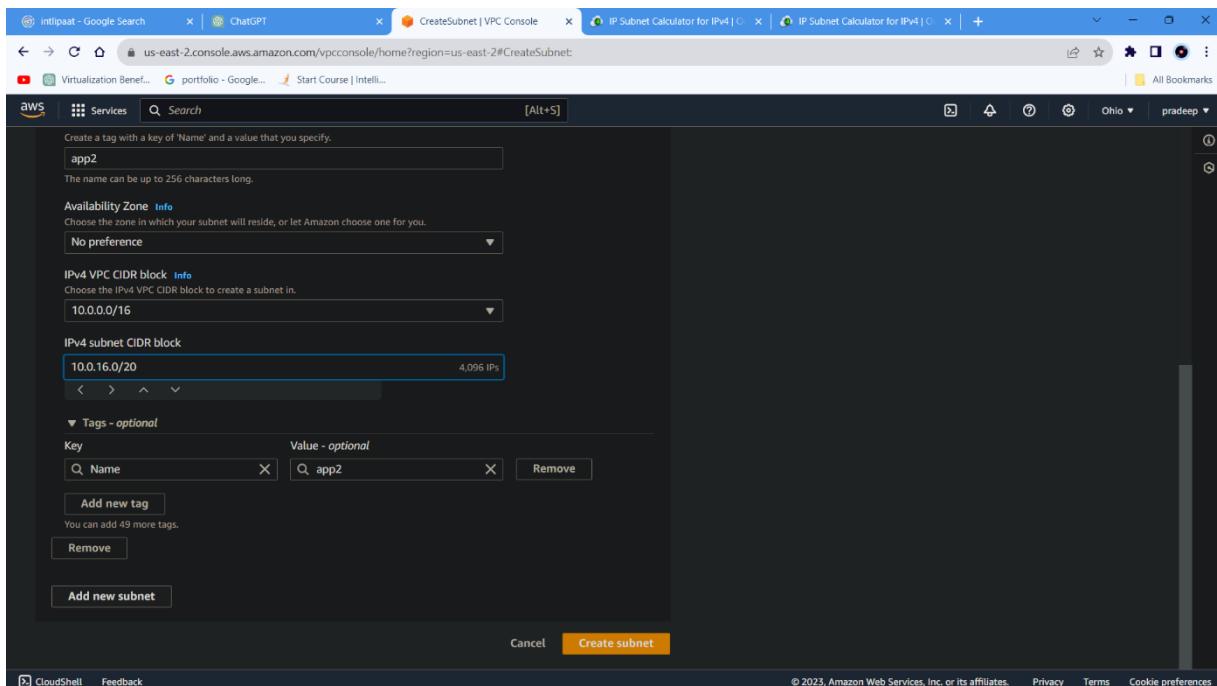
13.

The screenshot shows the "CreateSubnet" wizard, step 2. The "Name" field is set to "app1". Under "Availability Zone", "No preference" is selected. The "IPv4 VPC CIDR block" is set to "10.0.0.0/16". The "IPv4 subnet CIDR block" is set to "10.0.0.0/20". In the "Tags - optional" section, a tag "Name" is added with value "app1".

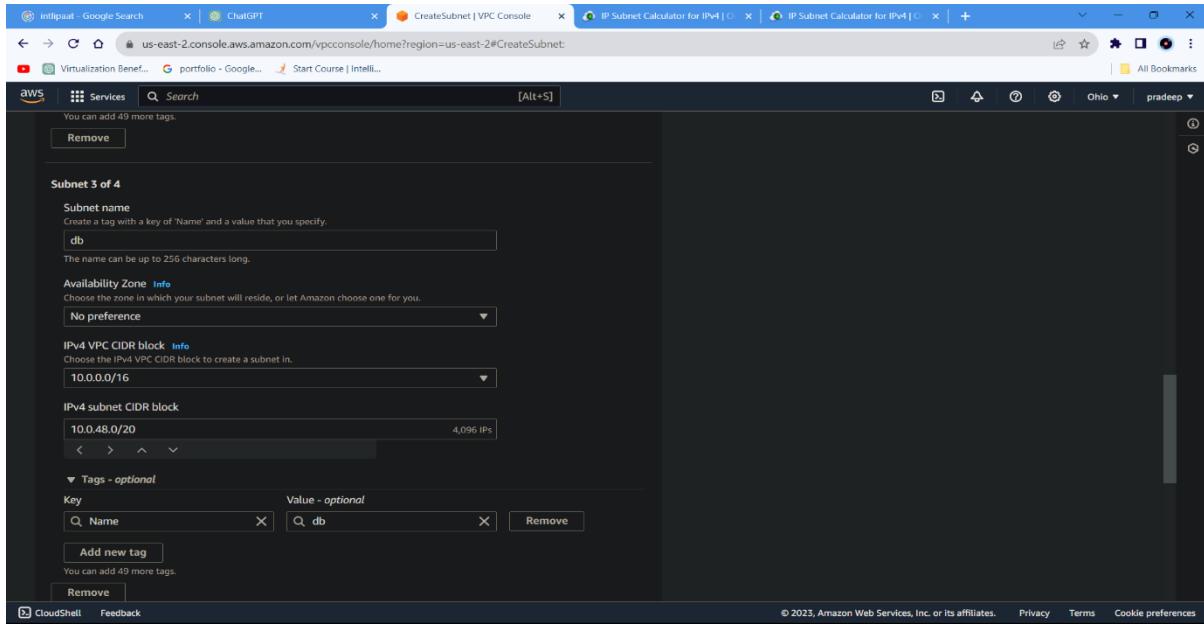
14.



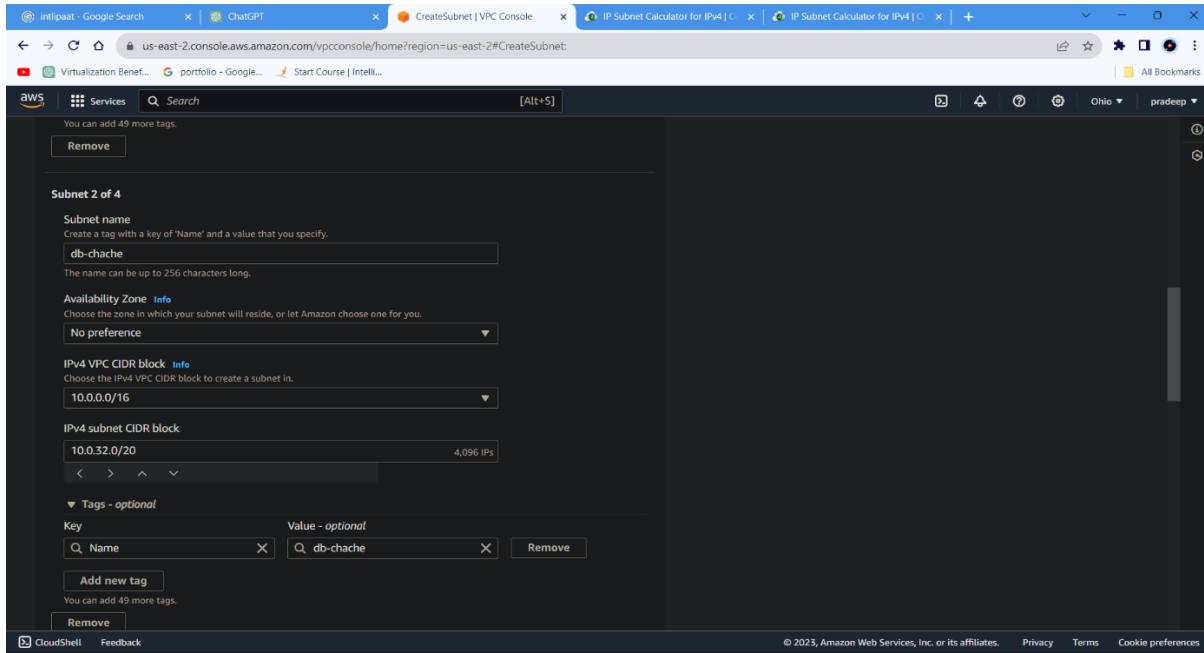
15.



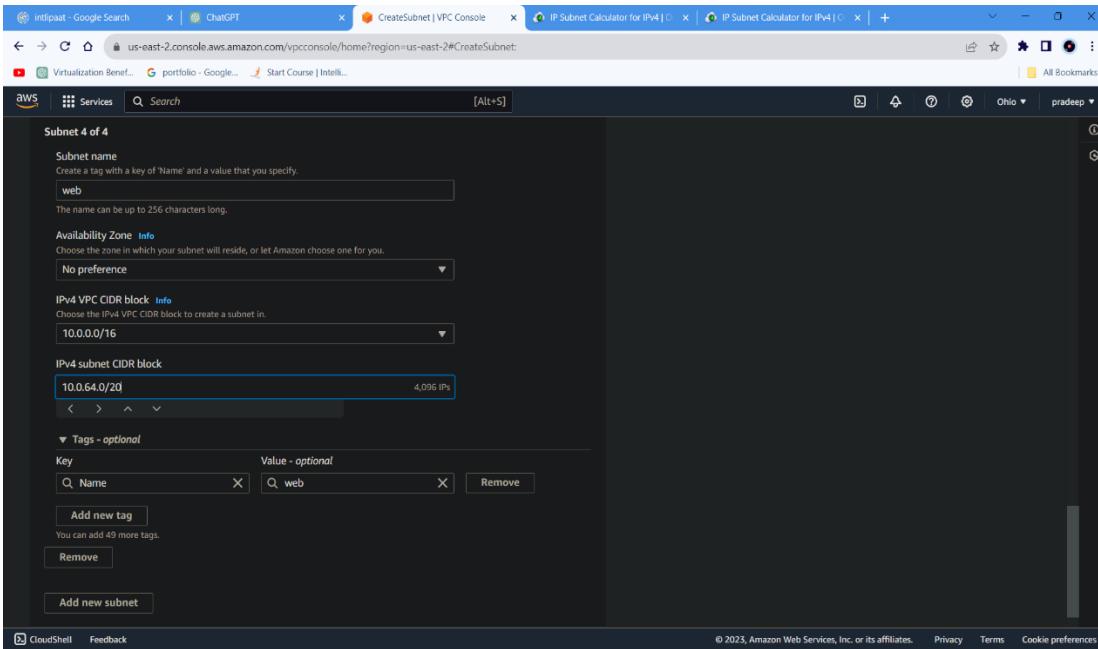
16.



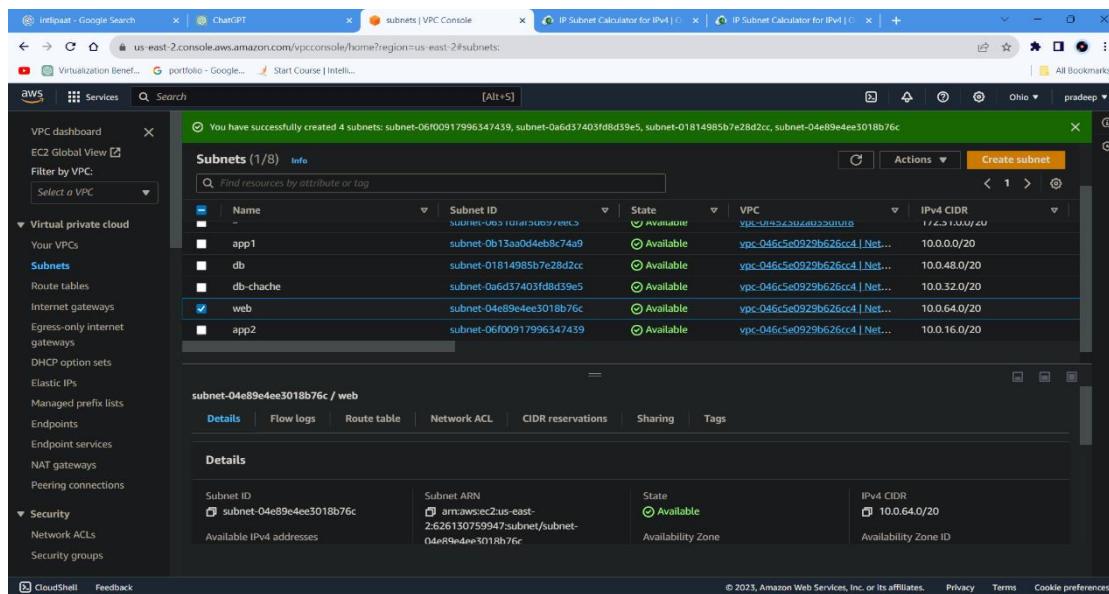
17.



18.



19.



20.

The screenshot shows the AWS VPC Subnets page. A success message at the top indicates 4 subnets have been created. The main table lists the following subnets:

Name	Subnet ID	State	VPC
subnets-00917996347439	subnet-0a6d37403d8d39e5	Available	vpc-046c5e0929b626cc4 Net
app1	subnet-0b13aa0d4eb8c74a9	Available	vpc-046c5e0929b626cc4 Net
db	subnet-01814985b7e28d2cc	Available	vpc-046c5e0929b626cc4 Net
db-cache	subnet-0a6d37403fd8d39e5	Available	vpc-046c5e0929b626cc4 Net
web	subnet-04e89e4ee3018b76c	Available	vpc-046c5e0929b626cc4 Net
app2	subnet-06f00917996347439	Available	vpc-046c5e0929b626cc4 Net

Actions for the selected subnet (web) include: View details, Create flow log, Edit subnet settings (highlighted), Edit IPv6 CIDRs, Edit network ACL association, Edit route table association, Edit CIDR reservations, Share subnet, Manage tags, and Delete subnet.

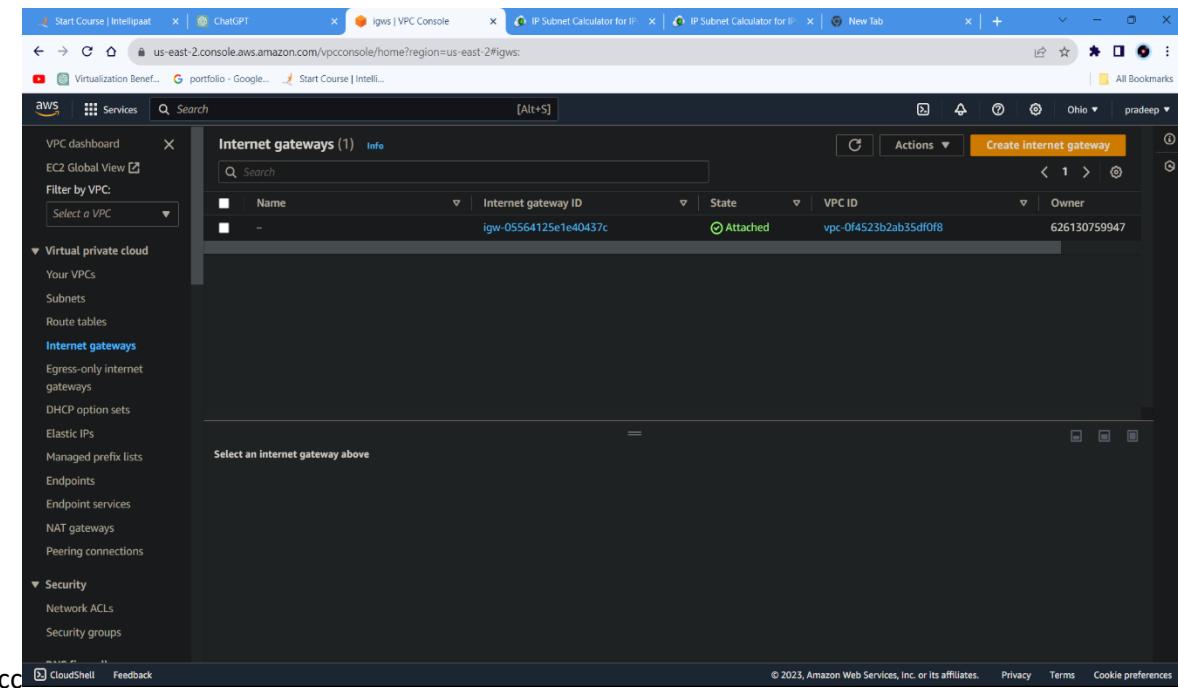
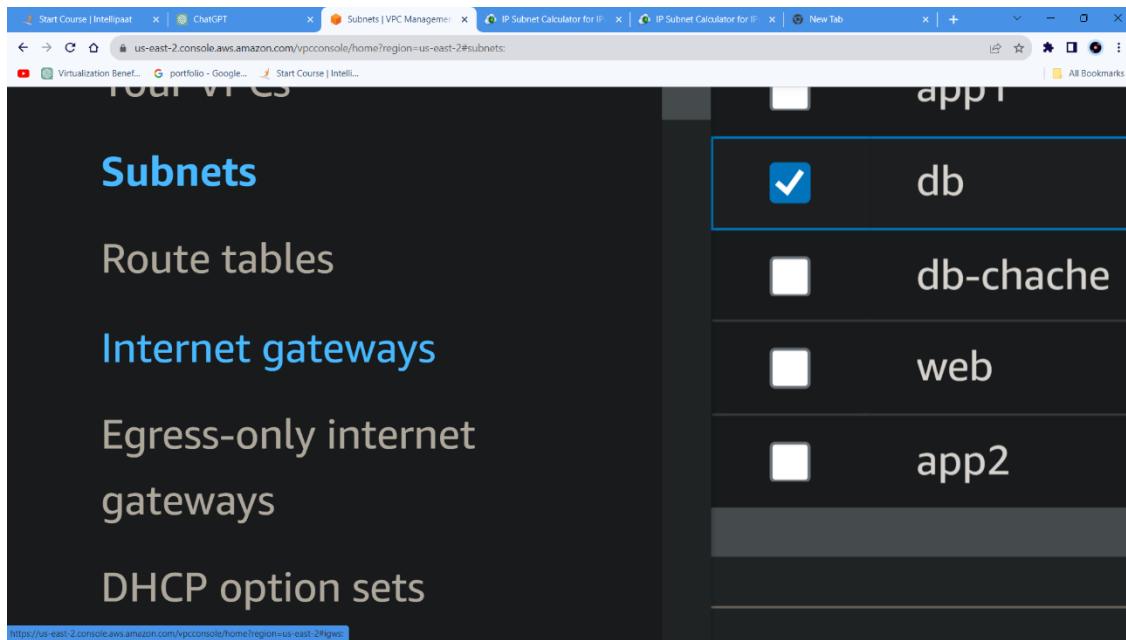
21.

The screenshot shows the 'EditSubnetSettings' dialog box for the subnet-04e89e4ee3018b76c. The configuration includes:

- Auto-assign IP settings:** Enable auto-assign public IPv4 address (checked).
- Resource-based name (RBN) settings:** Enable resource name DNS A record on launch (checked).
- DNS64 settings:** Enable DNS64 (checked).

At the bottom, there are 'Cancel' and 'Save' buttons.

lgw



The screenshot shows the 'Create internet gateway' wizard on the AWS VPC console. The current step is 'Internet gateway settings'. It includes fields for a 'Name tag' (set to 'network-production-igw') and a 'Tags - optional' section where a single tag ('Name: network-production-igw') is added. At the bottom are 'Cancel' and 'Create internet gateway' buttons.

The screenshot shows the 'InternetGateway | VPC' page on the AWS VPC console. A green banner at the top indicates that an internet gateway has been created. The main area displays the details of the new gateway, 'igw-0749bb8cf88dbcefa / network-production-igw'. The 'Details' tab is selected, showing the Internet gateway ID (igw-0749bb8cf88dbcefa), State (Detached), VPC ID (-), and Owner (626130759947). The 'Tags' section shows the tag 'Name: network-production-igw'. There is also a 'Actions' dropdown menu.

The screenshot shows the AWS VPC console with the 'Internet gateways' page. There are two gateways listed:

Name	Internet gateway ID	State	VPC ID
-	igw-05564125e1e40437c	Attached	vpc-0f4523
network-production-igw	igw-0749bb8cf88dbcefa	Detached	-

A dropdown menu is open over the 'Attach to VPC' button for the selected gateway ('network-production-igw'). The menu items are:

- View details
- Attach to VPC
- Detach from VPC
- Manage tags
- Delete Internet gateway

The screenshot shows the 'Attach to VPC' dialog box. The gateway 'igw-0749bb8cf88dbcefa' is selected. The 'Available VPCs' section shows one VPC:

VPC
vpc-046c5e0929b626cc4

A dropdown menu is open over the 'Attach internet gateway' button. The menu items are:

- AWS Command Line Interface command
- Cancel
- Attach internet gateway

The screenshot shows the AWS VPC console interface. The left sidebar is collapsed. The main content area displays the details of an Internet gateway named 'igw-0749bb8cf88dbcefa'. The gateway is successfully attached to a VPC with ID 'vpc-046c5e0929b626cc4' and is owned by user 'pradeep'. It has one tag: 'Name' with value 'network-production-igw'. The status is 'Attached'.

Internet gateway ID	State	VPC ID	Owner
igw-0749bb8cf88dbcefa	Attached	vpc-046c5e0929b626cc4 Network-production-vpc	pradeep

The screenshot shows the AWS VPC console interface. The left sidebar is collapsed. The main content area displays a list of route tables. There are three route tables listed:

Name	Route table ID	Explicit subnet associations	Main	VPC
-	rtb-0a44f83d9f1b19a2d	-	Yes	vpc-0f452b2ab35df0f8
-	rtb-09d3f5f523e268461	-	Yes	vpc-0a6057a365f2533c8
-	rtb-072954dc95ffa65cc	-	Yes	vpc-046c5e0929b626cc4

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="Name"/>	<input type="text" value="network-production-RT"/> X Remove

Add new tag
You can add 49 more tags.

Cancel Create route table

CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Updated routes for rtb-046ac920c29cd79ee / network-production-RT successfully

rtb-046ac920c29cd79ee / network-production-RT

Details Info

Route table ID <input type="text" value="rtb-046ac920c29cd79ee"/>	Main <input checked="" type="checkbox" value="No"/>	Explicit subnet associations -	Edge associations -
VPC <input type="text" value="vpc-046c5e0929b626cc4 Network-production-vpc"/>	Owner ID <input type="text" value="626130759947"/>		

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0749bb8cf88dbcefa	Active	No
10.0.0.0/16	local	Active	No

Actions Edit routes

CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows the 'Edit routes' page for a specific route table. The table has one entry:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Buttons at the bottom include 'Add route', 'Cancel', 'Preview', and 'Save changes'.

The screenshot shows the 'Edit routes' page for a specific route table. The table has two entries:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway igw-0749bb8cf88dbcef4	-	No

A 'Remove' button is visible next to the second entry. Buttons at the bottom include 'Add route', 'Cancel', 'Preview', and 'Save changes'.

The screenshot shows the AWS VPC console interface. On the left, there's a navigation sidebar with options like 'VPC dashboard', 'EC2 Global View', 'Route tables' (which is selected), and 'Security'. The main content area displays a success message: 'Route table rtb-046ac920c29cd79ee | network-production-RT was created successfully.' Below this, it shows the details of the newly created route table, including its ID (rtb-046ac920c29cd79ee), VPC (vpc-046c5e0929b626cc4), and owner (626130759947). The 'Routes' tab is selected, showing one route entry: Destination 10.0.0.0/16, Target local, Status Active, and Propagated No. At the bottom right of the main content area, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2023, Amazon Web Services, Inc. or its affiliates.' and 'Cookie preferences'.

This screenshot shows the AWS VPC console with the 'Route tables' list page. The sidebar remains the same. The main area lists four route tables: 'rtb-044f83d9f1b19a2d', 'rtb-09d3f5523e268461', 'rtb-072954dc95ffae65cc', and 'network-production-RT' (which is selected). A context menu is open over the selected route table, with the 'Edit subnet associations' option highlighted. Below the list, it shows the details of the selected route table (rtb-046ac920c29cd79ee), which matches the one shown in the first screenshot. The bottom right of the main content area includes the usual footer links and copyright notice.

The screenshot shows the AWS VPC Route Table Subnet Associations editor. The URL in the browser is `us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#EditRouteTableSubnetAssociations:RouteTableId=rtb-046ac920c29cd79ee`. The page title is "Edit subnet associations".

Available subnets (1/5)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
db	subnet-01814985b7e28d2cc	10.0.48.0/20	-	Main (rtb-072954dc95ffa65cc)
db-chache	subnet-0a6d357403fd8d359e5	10.0.32.0/20	-	Main (rtb-072954dc95ffa65cc)
app1	subnet-0b13aa0d4eb8c74a9	10.0.0.0/20	-	Main (rtb-072954dc95ffa65cc)
<input checked="" type="checkbox"/> web	subnet-04e89e4ee5018b76c	10.0.64.0/20	-	Main (rtb-072954dc95ffa65cc)
app2	subnet-06f00917996347439	10.0.16.0/20	-	Main (rtb-072954dc95ffa65cc)

Selected subnets

subnet-04e89e4ee5018b76c / web

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Instance.

22.

The screenshot shows the AWS EC2 Dashboard for the US East (Ohio) Region. The left sidebar includes sections for Instances, Images, Elastic Block Store, and Network & Security. The main area displays EC2 Global View statistics and a Launch instance section. A sidebar on the right provides EC2 Free Tier Info, showing 0 offers for all AWS Regions, and details about EC2 free tier offers in use, which have exceeded the free tier limit. It also shows offer usage monthly for Linux EC2 Instances and Storage space on EBS.

23.

The screenshot shows the 'Launch an instance' wizard. In the 'Name and tags' step, the name 'app1' is entered. In the 'Application and OS Images (Amazon Machine Image)' step, the Amazon Linux 2023 AMI 2023.2.2 is selected. The 'Summary' step shows the configuration: 1 instance, t2.micro instance type, New security group, and 1 volume(s) - 8 GiB storage. A tooltip for the Free tier indicates it includes 750 hours of t2.micro (or t3.micro in regions where t2.micro is unavailable) instance usage on free tier AMIs per year. The 'Launch instance' button is highlighted.

24.

The screenshot shows the AWS Lambda console with the following details:

- Region:** us-east-2
- Function name:** Lambda function
- Description:** This function processes file uploads from an S3 bucket and triggers a scheduled event every 15 minutes.
- Runtime:** Python 3.8
- Handler:** lambda_function.lambda_handler
- Memory size:** 128 MB
- Timeout:** 300 seconds
- Code:** zip file uploaded from local machine.
- Environment:** No environment variables set.
- Logs:** CloudWatch Logs stream created: /aws/lambda/Lambda function
- Deployment:** Deployment package uploaded: lambda_function.zip
- Logs:** CloudWatch Logs stream created: /aws/lambda/Lambda function

25.

The screenshot shows the AWS Lambda console with the following details:

- Region:** us-east-2
- Function name:** Lambda function
- Description:** This function processes file uploads from an S3 bucket and triggers a scheduled event every 15 minutes.
- Runtime:** Python 3.8
- Handler:** lambda_function.lambda_handler
- Memory size:** 128 MB
- Timeout:** 300 seconds
- Code:** zip file uploaded from local machine.
- Environment:** No environment variables set.
- Logs:** CloudWatch Logs stream created: /aws/lambda/Lambda function
- Deployment:** Deployment package uploaded: lambda_function.zip
- Logs:** CloudWatch Logs stream created: /aws/lambda/Lambda function

26.

The screenshot shows the AWS Lambda console with the following details:

- Function name:** Lambda function
- Runtime:** Python 3.9
- Handler:** lambda_function.lambda_handler
- Memory size:** 128 MB
- Timeout:** 3 seconds
- Code:** zip file (1.00 KB)
- Environment:** None
- Logs:** CloudWatch Logs
- Deployment:** None
- Logs:** CloudWatch Logs

At the bottom, there is a "Create function" button.

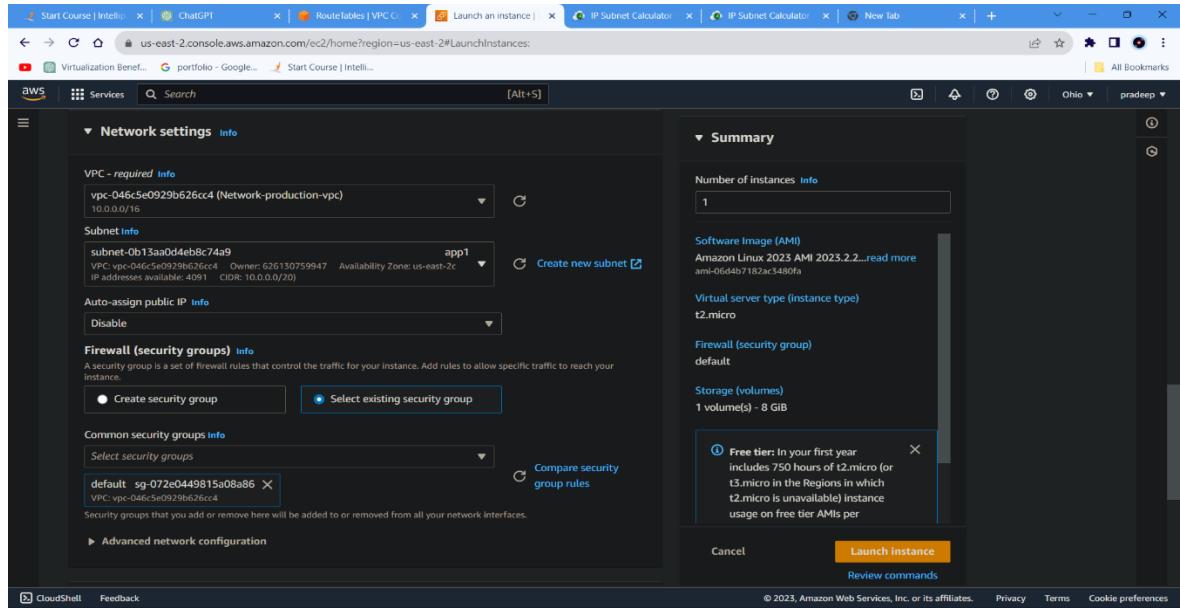
27.

The screenshot shows the AWS Lambda console with the following details:

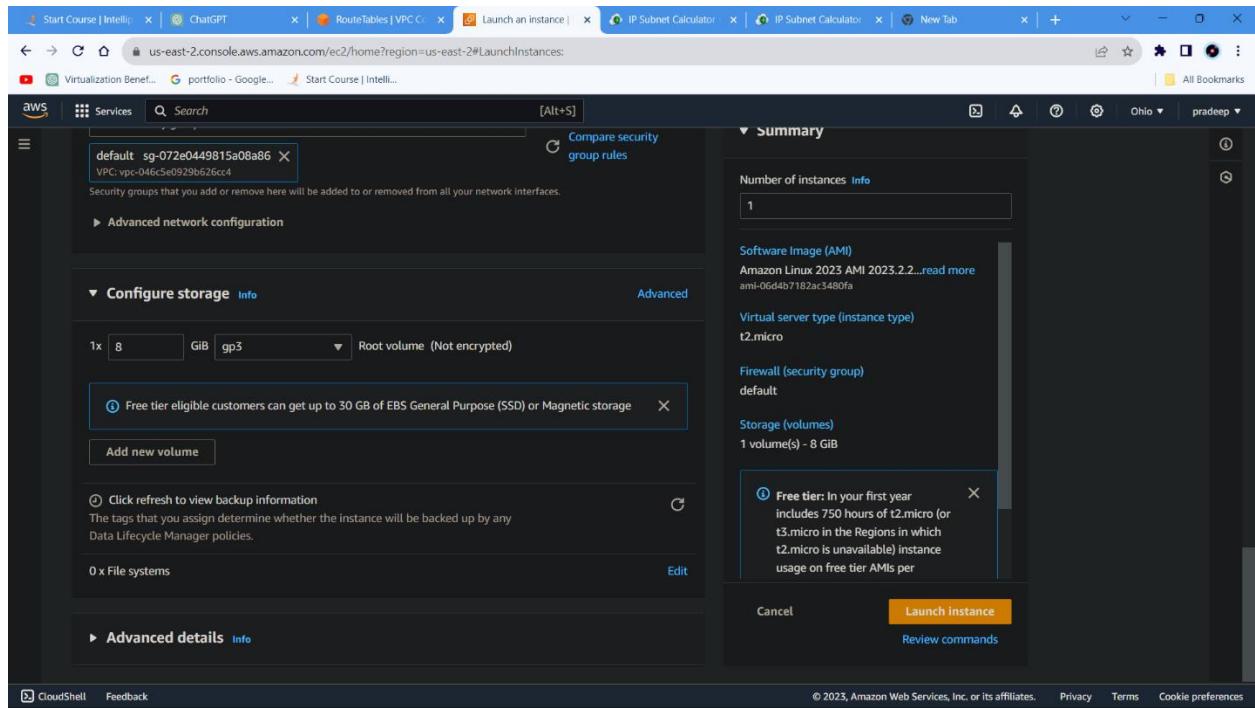
- Function name:** Lambda function
- Runtime:** Python 3.9
- Handler:** lambda_function.lambda_handler
- Memory size:** 128 MB
- Timeout:** 3 seconds
- Code:** zip file (1.00 KB)
- Environment:** None
- Logs:** CloudWatch Logs
- Deployment:** None
- Logs:** CloudWatch Logs

At the bottom, there is a "Create function" button.

28.



29.



30.

The screenshot shows the AWS EC2 Instances page. The left sidebar is collapsed. The main area displays a table titled "Instances (1) Info" with one row. The row contains the following information:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
app1	i-0aa7cf3d1fb8ceef	Pending	t2.micro	-	No alarms	us-east-2c	-

A modal window titled "Select an instance" is open at the bottom of the screen, listing "app1" as the only option.

31.

The screenshot shows the "Launch an instance" wizard. The left sidebar shows the navigation path: EC2 > Instances > Launch an instance. The main area is divided into several sections:

- Launch an instance**: A brief introduction to creating instances.
- Name and tags**: A section where the instance name is set to "app2".
- Application and OS Images (Amazon Machine Image)**: A section describing AMIs and providing a search bar.
- Summary**: A summary panel on the right containing the following details:
 - Number of instances: 1
 - Software Image (AMI): Amazon Linux 2023.2.2... (with a "read more" link)
 - Virtual server type (instance type): t2.micro
 - Firewall (security group): New security group
 - Storage (volumes): 1 volume(s) - 8 GB
- Free tier**: A callout box explaining the free tier usage for the first year.
- Buttons**: "Launch instance" (highlighted in orange), "Cancel", and "Review commands".

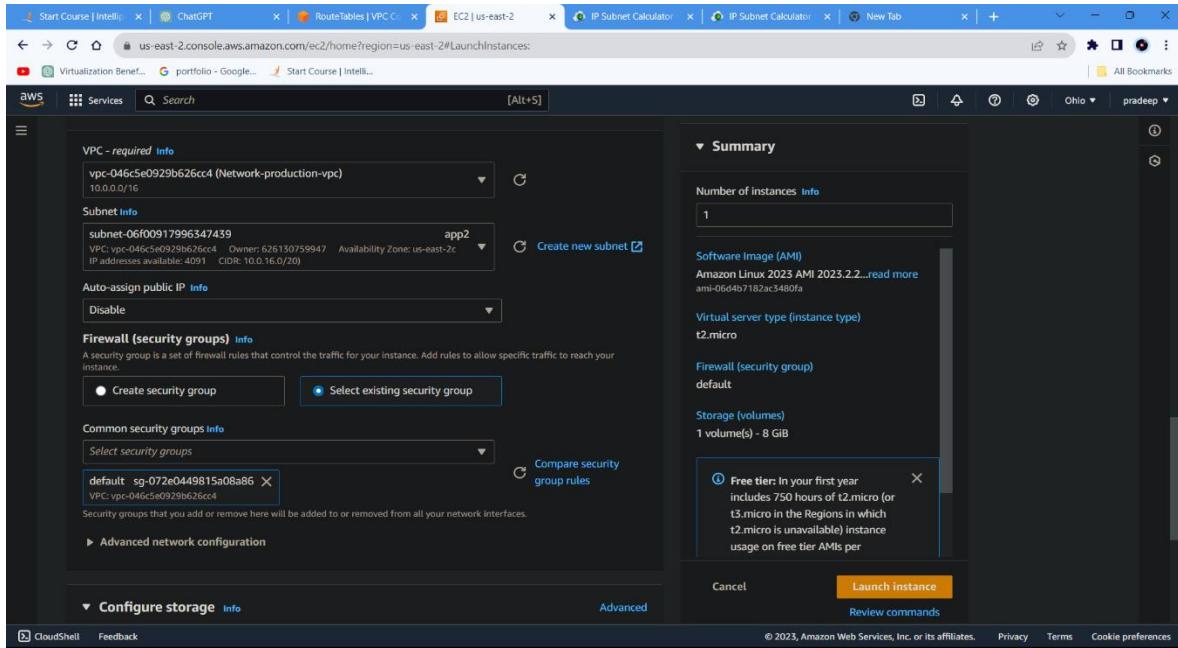
32.

The screenshot shows the AWS EC2 Launch Instances page. In the search bar at the top, the name "app" is entered. Below the search bar, the "Application and OS Images (Amazon Machine Image)" section is expanded, showing a list of available AMIs. One item, "Amazon Linux 2023 AMI", is selected and highlighted with a blue border. This item includes details such as the AMI ID (ami-06d4b7182ca5480fa), Virtualization type (hvm), and ENA support (true). To the right of the AMI list, there is a summary section showing 1 instance, the software image (Amazon Linux 2023 AMI 2023.2.2...), the virtual server type (t2.micro), and storage (1 volume(s) - 8 GiB). A tooltip for the free tier is displayed, stating: "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 year." At the bottom right, there are "Launch Instance" and "Review commands" buttons.

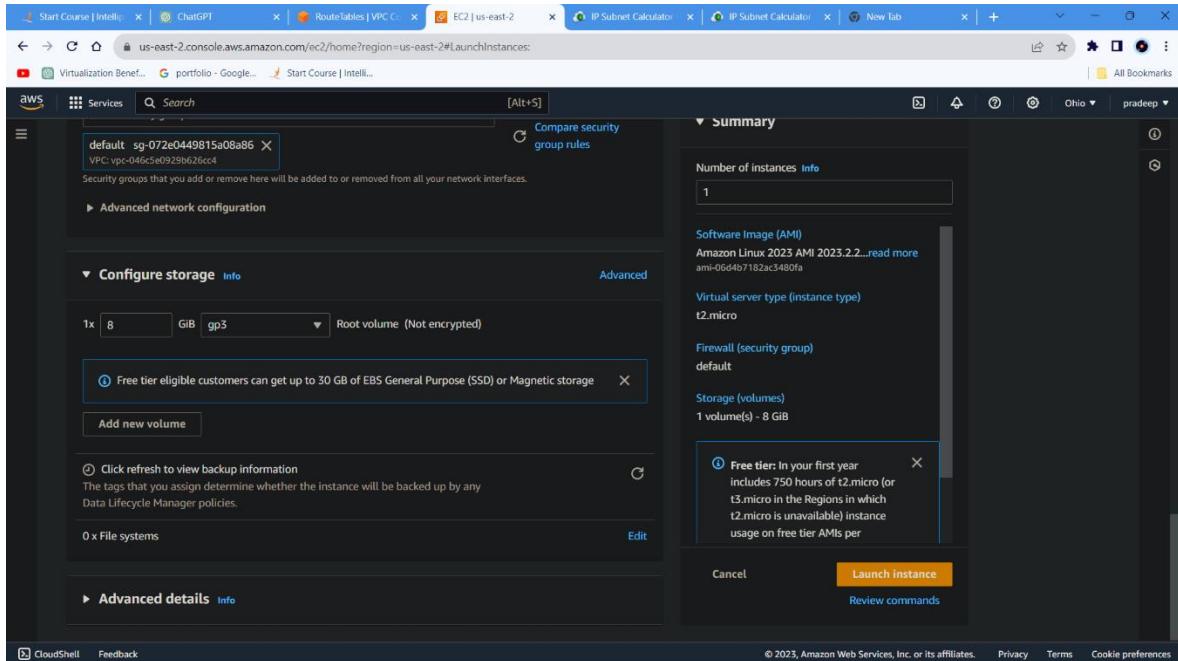
33.

The screenshot shows the AWS EC2 Launch Instances page. The "Instance type" section is expanded, showing the "t2.micro" option selected. Below this, a note states: "Additional costs apply for AMIs with pre-installed software". The "Key pair (login)" section is expanded, showing a key pair named "newkey" selected. The "Network settings" section is expanded, showing a network interface with the ID "vpc-0f4523b2ab35df0f0". On the right side, the "Summary" section shows 1 instance, the software image (Amazon Linux 2023 AMI 2023.2.2...), the virtual server type (t2.micro), and storage (1 volume(s) - 8 GiB). A tooltip for the free tier is displayed, stating: "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 year." At the bottom right, there are "Launch Instance" and "Review commands" buttons.

34.



35.



36.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The 'Name and tags' section has 'db-chache' entered in the Name field. The 'Application and OS Images (Amazon Machine Image)' section shows a search bar and a list of OS options: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux. A tooltip for the 'Free tier' is visible, stating: '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'. The 'Launch instance' button is at the bottom right.

37.

The screenshot shows the 'Launch an instance' wizard with more detailed configuration. Under 'Amazon Machine Image (AMI)', the 'Amazon Linux 2023 AMI' is selected. It shows the AMI ID: ami-06d4b7182ac3480fa, Architecture: 64-bit (x86), and AMI ID: ami-0090be1905998682a. A tooltip for the 'Free tier' is visible, stating: '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'. The 'Launch instance' button is at the bottom right.

38.

The screenshot shows the AWS EC2 Launch Instances wizard. In the 'Instance type' section, a t2.micro instance is selected. The summary panel indicates 1 instance, using the Amazon Linux 2023 AMI, t2.micro instance type, and a new security group. A tooltip for the free tier shows it includes 750 hours of usage per year. The 'Launch instance' button is highlighted.

39.

The screenshot shows the AWS EC2 Launch Instances wizard. In the 'Key pair (login)' section, a new key pair named 'newkey' is selected. In the 'Network settings' section, the VPC 'vpc-0f4523b2ab35df0f8' and subnet 'No preference (Default subnet in any availability zone)' are chosen. The summary panel indicates 1 instance, using the Amazon Linux 2023 AMI, t2.micro instance type, and a new security group. A tooltip for the free tier shows it includes 750 hours of usage per year. The 'Launch instance' button is highlighted.

40.

The screenshot shows the AWS EC2 Launch Instances wizard. On the left, under 'Network settings', it lists a VPC (vpc-046c5e0929b626cc4) and a subnet (subnet-0a6d37405fd8d39e5). It also shows an 'Auto-assign public IP' dropdown set to 'Disable'. Under 'Firewall (security groups)', there are two options: 'Create security group' (selected) and 'Select existing security group'. A security group 'default sg-072e0449815a08a86' is selected. On the right, the 'Summary' section shows 1 instance, the AMI (Amazon Linux 2023.2.2...), the instance type (t2.micro), and storage (1 volume(s) - 8 GiB). A modal window displays the 'Free tier: In your first year' offer, which includes 750 hours of t2.micro or t3.micro usage per year. At the bottom, there are 'Cancel', 'Launch instance', and 'Review commands' buttons.

41.

The screenshot shows the AWS EC2 Launch Instances wizard. On the left, under 'Configure storage', it shows a root volume of 8 GiB using gp3 storage. A note indicates that free-tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Below this, there's a section for file systems and a note to click refresh to view backup information. On the right, the 'Summary' section shows 1 instance, the AMI (Amazon Linux 2023.2.2...), the instance type (t2.micro), and storage (1 volume(s) - 8 GiB). A modal window displays the 'Free tier: In your first year' offer, which includes 750 hours of t2.micro or t3.micro usage per year. At the bottom, there are 'Cancel', 'Launch instance', and 'Review commands' buttons.

40.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The 'Summary' step is selected. Configuration details include:

- Name and tags**: Name is set to 'app1'.
- Software Image (AMI)**: Amazon Linux 2023 AMI 2023.2.2... (ami-06d4b7182ac3480fa).
- Virtual server type (instance type)**: t2.micro.
- Firewall (security group)**: New security group.
- Storage (volumes)**: 1 volume(s) - 8 GiB.

A tooltip for the 'Free tier' is visible, stating: "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".

At the bottom right are 'Cancel', 'Launch instance' (highlighted in orange), and 'Review commands' buttons.

41.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console, currently on the 'Amazon Machine Image (AMI)' step. The 'Amazon Linux 2023 AMI' is selected. Key details shown are:

- Description**: Amazon Linux 2023 AMI 2023.2.20231113.0.x86_64 HVM kernel-6.1.
- Architecture**: 64-bit (x86).
- AMI ID**: ami-06d4b7182ac3480fa.
- Verified provider**: A green button.

A tooltip for the 'Free tier' is visible, stating: "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".

At the bottom right are 'Cancel', 'Launch instance' (highlighted in orange), and 'Review commands' buttons.

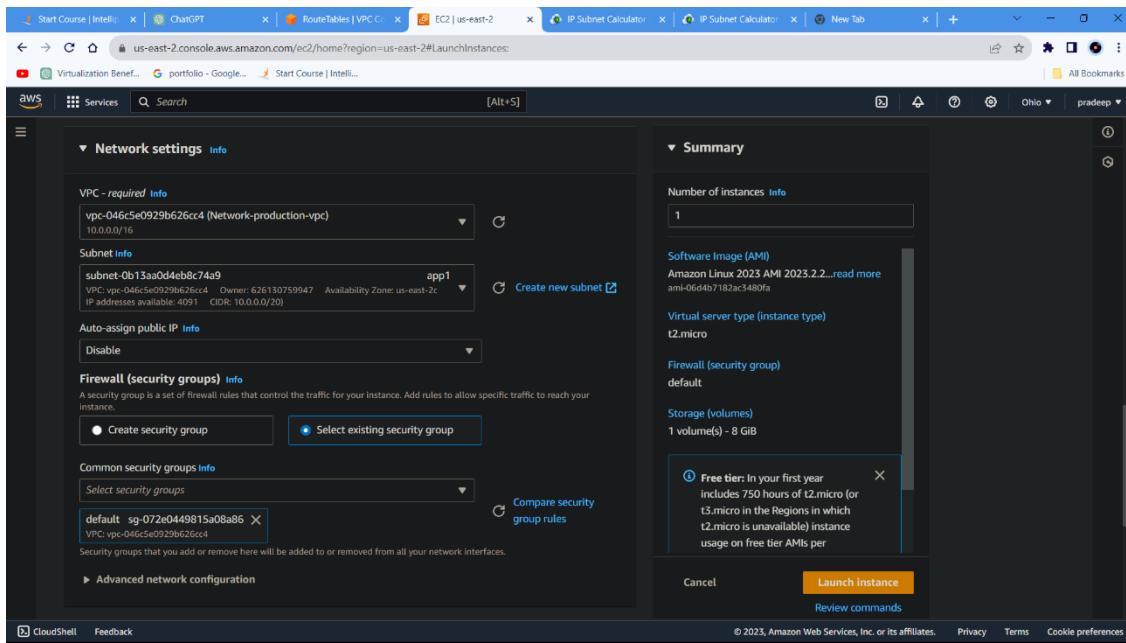
42.

The screenshot shows the AWS EC2 Launch Instances page. The instance type selected is t2.micro, which is free tier eligible. The software image (AMI) is set to Amazon Linux 2023.2.2. The virtual server type is t2.micro. A tooltip for the free tier indicates it includes 750 hours of t2.micro usage per year. The storage volume is 1 volume(s) - 8 GiB. The summary section shows 1 instance. The launch button is visible at the bottom right.

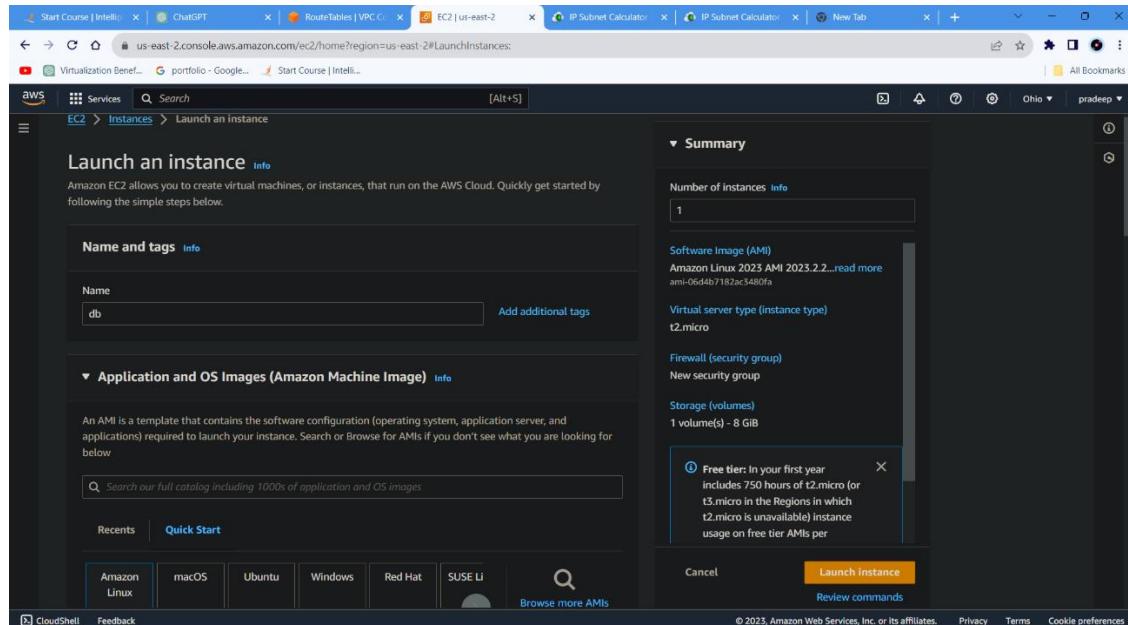
43.

The screenshot shows the AWS EC2 Launch Instances page. A new key pair named "newkey" is selected. In the network settings, the subnet is set to "No preference (Default subnet in any availability zone)". Under Firewall (security groups), a new security group named "launch-wizard-22" is being created with a rule allowing SSH traffic from anywhere. The summary section shows 1 instance. The launch button is visible at the bottom right.

44.



45



46.

The screenshot shows the AWS EC2 Launch Instances wizard. In Step 1: Choose AMI, the user has selected the Amazon Linux 2023 AMI. The summary section indicates 1 instance, showing the AMI details: Amazon Linux 2023 AMI (ami-06d4b7182ac3480fa), Free tier eligible, 64-bit (x86) architecture, and AMI ID ami-06d4b7182ac3480fa. The instance type is set to t2.micro, and the storage is 1 volume(s) - 8 GiB. A tooltip for the free tier is displayed, stating: "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".

47.

The screenshot shows the AWS EC2 Launch Instances wizard. In Step 1: Choose AMI, the user has selected the Amazon Linux 2023 AMI. The summary section indicates 1 instance, showing the AMI details: Amazon Linux 2023 AMI (ami-06d4b7182ac3480fa), Free tier eligible, t2.micro instance type, and 8 GiB storage. A tooltip for the free tier is displayed, stating: "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".

48.

The screenshot shows the AWS EC2 Launch Instances wizard. On the left, under 'Network settings', there's a 'VPC - required' section with a dropdown set to 'vpc-046c5e0929b626cc4 (Network-production-vpc)'. Below it is a 'Subnet Info' section with a dropdown for 'subnet-0181498b7e28d2cc' and a link to 'Create new subnet'. Under 'Auto-assign public IP', the 'Disable' option is selected. In the 'Firewall (security groups)' section, the 'Select existing security group' radio button is selected, and a dropdown shows 'default sg-072e0449815a08a86'. A tooltip for this dropdown indicates it includes 'VPC: vpc-046c5e0929b626cc4'. The 'Common security groups' section shows 'Select security groups' with 'default' selected. A tooltip for 'Compare security group rules' is visible. On the right, the 'Summary' section shows 'Number of instances: 1'. It also lists 'Software Image (AMI): Amazon Linux 2023 AMI 2023.2.2...', 'Virtual server type (instance type): t2.micro', 'Firewall (security group): default', and 'Storage (volumes): 1 volume(s) - 8 GiB'. A tooltip for 'Free tier' is shown, stating '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'. At the bottom are 'Cancel', 'Launch instance', and 'Review commands' buttons.

49.

The screenshot shows the AWS EC2 Launch Instances wizard. On the left, under 'Configure storage', there's a dropdown for 'Root volume (Not encrypted)' set to '1x 8 GiB gp3'. A tooltip for this dropdown indicates it includes 'VPC: vpc-046c5e0929b626cc4'. Below it is a 'Free tier' message: 'Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage'. An 'Add new volume' button is present. Under 'Advanced details', there's a 'File systems' section with a '0 x File systems' dropdown and an 'Edit' button. On the right, the 'Summary' section shows 'Number of instances: 1'. It also lists 'Software Image (AMI): Amazon Linux 2023 AMI 2023.2.2...', 'Virtual server type (instance type): t2.micro', 'Firewall (security group): default', and 'Storage (volumes): 1 volume(s) - 8 GiB'. A tooltip for 'Free tier' is shown, stating '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'. At the bottom are 'Cancel', 'Launch instance', and 'Review commands' buttons.

50.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Name and tags'. A modal window titled 'Summary' is open, showing the configuration details. The 'Software Image (AMI)' is set to 'Amazon Linux 2023 AMI 2023.2.2...'. The 'Virtual server type (instance type)' is 't2.micro'. The 'Storage (volumes)' section shows '1 volume(s) - 8 GiB'. A tooltip for the 'Free tier' indicates it includes 750 hours of t2.micro usage per year. The 'Launch instance' button is visible at the bottom right of the modal.

51.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Application and OS Images (Amazon Machine Image)'. A modal window titled 'Summary' is open, showing the configuration details. The 'Software Image (AMI)' is set to 'Amazon Linux 2023 AMI 2023.2.2...'. The 'Virtual server type (instance type)' is 't2.micro'. The 'Storage (volumes)' section shows '1 volume(s) - 8 GiB'. A tooltip for the 'Free tier' indicates it includes 750 hours of t2.micro usage per year. The 'Launch instance' button is visible at the bottom right of the modal.

52.

Start Course | Intelli... ChatGPT RouteTables | VPC C EC2 | us-east-2 IP Subnet Calculator IP Subnet Calculator New Tab All Bookmarks Virtualization Benefit... portfolio - Google... Start Course | Intelli... Services Search [Alt+S]

Summary

Number of instances **1**

Software Image (AMI)
Amazon Linux 2023 AMI 2023.2.2...read more
ami-06d4b7182ac3480fa

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 year

Cancel Launch instance Review commands

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

53.

Start Course | Intelli... ChatGPT RouteTables | VPC C EC2 | us-east-2 IP Subnet Calculator IP Subnet Calculator New Tab All Bookmarks Virtualization Benefit... portfolio - Google... Start Course | Intelli... Services Search [Alt+S]

Summary

Number of instances **1**

Software Image (AMI)
Amazon Linux 2023 AMI 2023.2.2...read more
ami-06d4b7182ac3480fa

Virtual server type (instance type)
t2.micro

Firewall (security group)
default

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 year

Cancel Launch instance Review commands

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

54.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations (New), Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security. The main area displays a table titled 'Instances (6) Info' with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. The instances listed are db-cache, app1, db, no, and app2, all in the 'Running' state. Below the table, a modal window titled 'Select an instance' is open, showing a list of instances: db-cache, app1, db, no, and app2. At the bottom right of the main area, there are links for CloudShell and Feedback, and a footer with copyright information and links for Privacy, Terms, and Cookie preferences.

55.

The screenshot shows the AWS EC2 Security Groups page. The URL in the address bar is 'us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#ModifyInboundSecurityGroupRules:securityGroupId=sg-072e0449815a08a86'. The main content area is titled 'Edit inbound rules' with a 'Info' link. It says 'Inbound rules control the incoming traffic that's allowed to reach the instance.' Below this is a table titled 'Inbound rules info' with columns: Security group rule ID, Type info, Protocol info, Port range info, Source info, and Description - optional info. A single row is present with a 'Delete' button. At the bottom, there's a note: '⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' There are 'Cancel', 'Preview changes', and 'Save rules' buttons. The footer includes links for CloudShell and Feedback, and standard AWS footer links for Privacy, Terms, and Cookie preferences.

56.

The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 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 displays a table of instances:

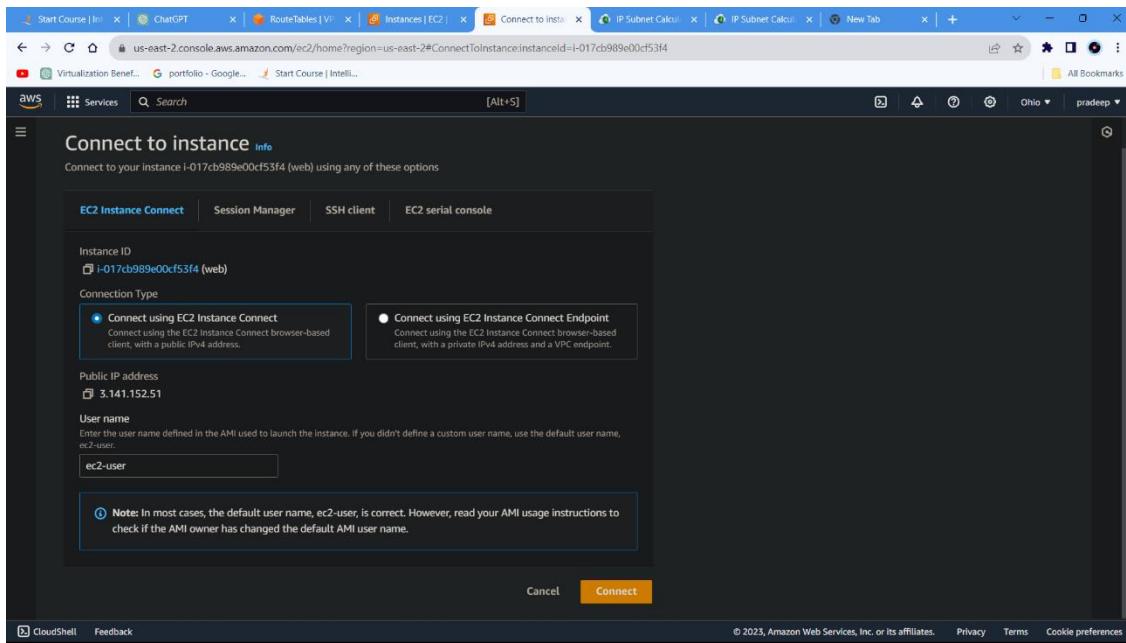
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
db-chache	i-0e6afce19add716e	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
app1	i-00d90e3ce9c2cf79b	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
db	i-0ccfb86cbc1fa144	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
web	i-017cb989e00cf53f4	Running	t2.micro	Initializing	No alarms	us-east-2c	-
app2	i-0b1f0945a7af737b5	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-

57.

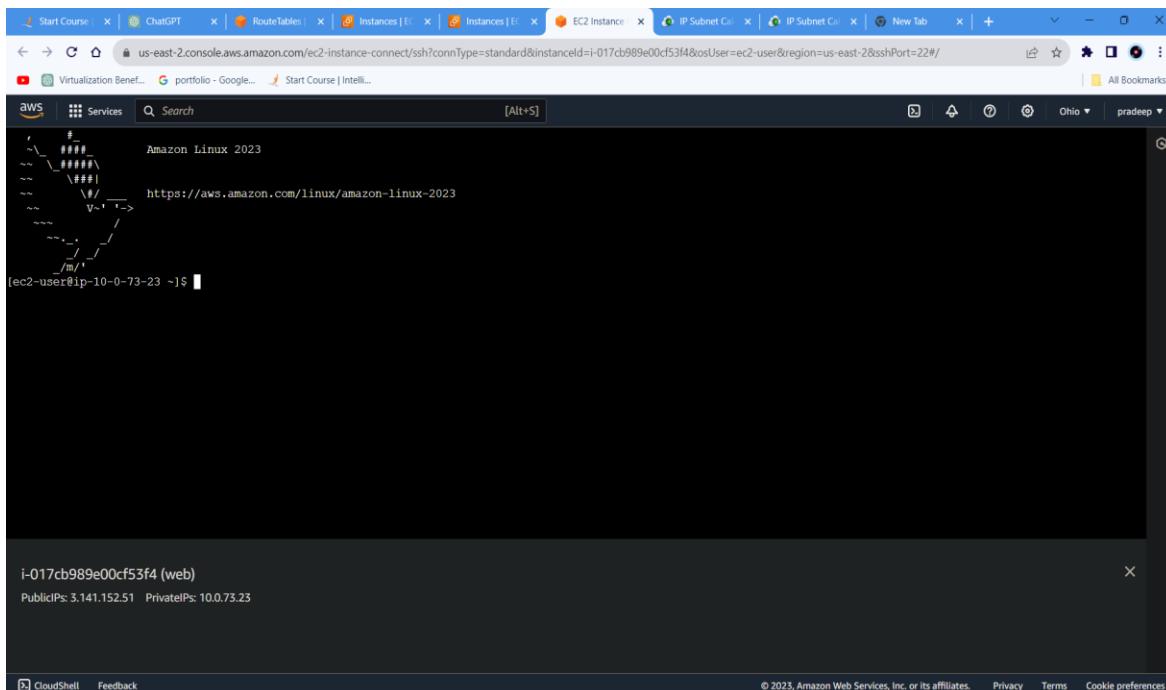
The screenshot shows the AWS EC2 Instances page, similar to the previous one, but with the 'web' instance selected. The main content area displays detailed information for the 'web' instance:

Details		Security	Networking	Storage	Status checks	Monitoring	Tags
Security details							
IAM Role							
-							
Owner ID							
626130759947							
Launch time							
Sun Dec 03 2023 15:24:28 GMT+0530 (India Standard Time)							

58.



59.



60.

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec  3 10:01:33 2023 from 3.16.146.3
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.250.190.110) 56(84) bytes of data.
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=1 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=2 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=3 ttl=108 time=16.7 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 16.656/16.696/16.725/0.029 ms
[ec2-user@ip-10-0-73-23 ~]$
```

i-017cb989e00cf53f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23

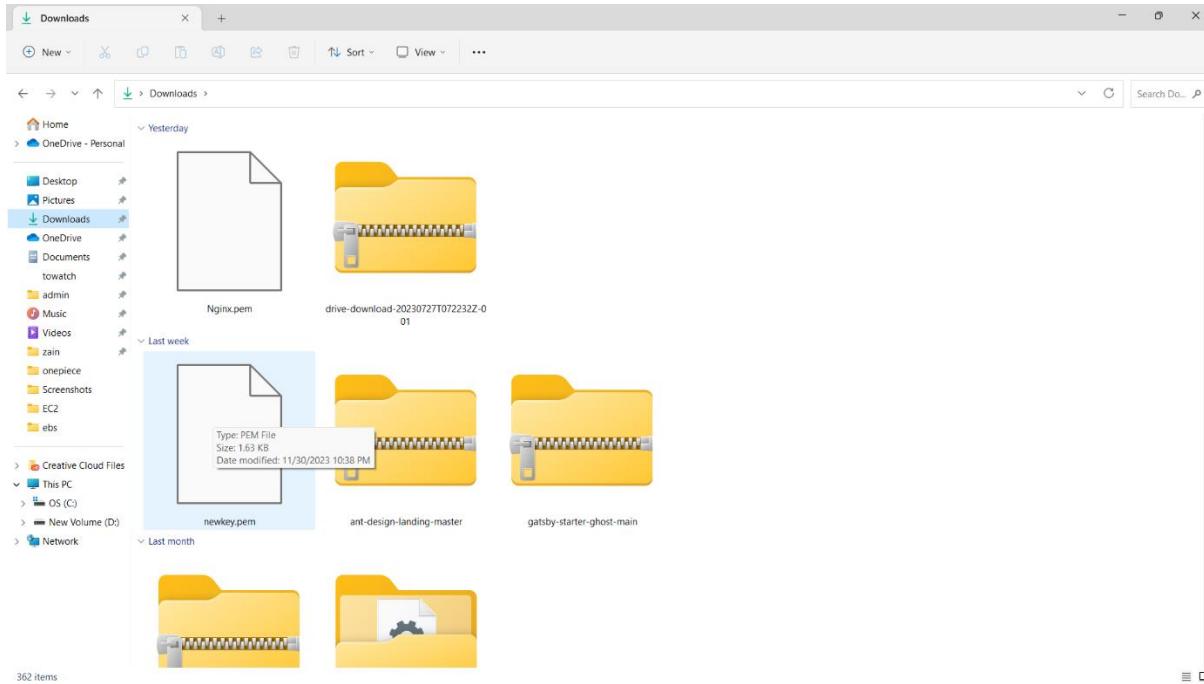
61.

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

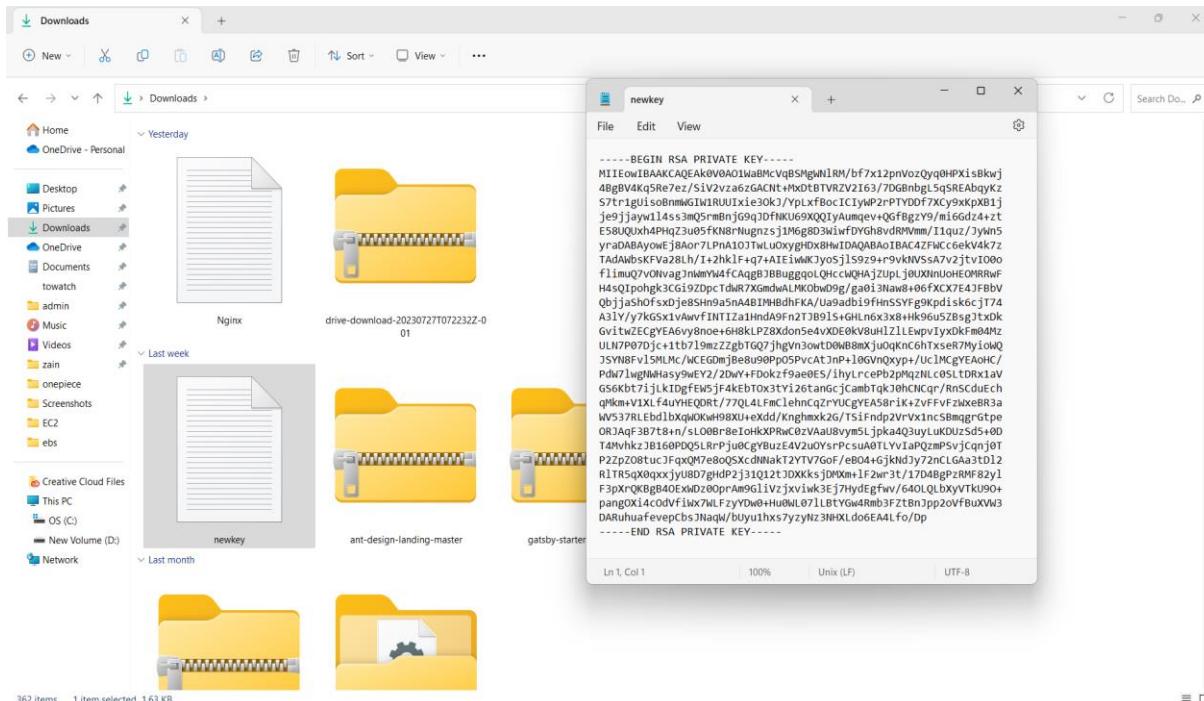
Last login: Sun Dec  3 10:01:33 2023 from 3.16.146.3
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.250.190.110) 56(84) bytes of data.
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=1 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=2 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=3 ttl=108 time=16.7 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 16.656/16.696/16.725/0.029 ms
[ec2-user@ip-10-0-73-23 ~]$ sudo nano key.pem
```

i-017cb989e00cf53f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23

62.



63.



64.

The screenshot shows the AWS EC2 Instances page. On the left, the navigation menu includes EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, CloudShell, and Feedback. The main content area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
web	i-017cb989e00cf53f4	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
db-chache	i-0e6afce19add716e	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
app1	i-00d90e3ce9c2cf79b	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
db	i-0ccfb86cbcf1fa144	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-
no	i-0aa7cf3d1fb8ceef	Terminated	t2.micro	-	No alarms	us-east-2c	-
app2	i-0b1f0945a7af737b5	Running	t2.micro	2/2 checks passed	No alarms	us-east-2c	-

Below the table, a specific instance is selected: **Instance: i-00d90e3ce9c2cf79b (app1)**. The details tab is active, showing:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-00d90e3ce9c2cf79b (app1)	-	10.0.14.126
IPv6 address	Instance state	Public IPv4 DNS
-	Running	-
Hostname type	Private IP DNS name (IPv4 only)	Public IPv4 DNS
IP name: ip-10-0-14-126.us-east-2.compute.internal	ip-10-0-14-126.us-east-2.compute.internal	-

65.

The screenshot shows the AWS EC2 Instances page, identical to the previous one but with a different focus. The 'Actions' dropdown menu is open, highlighting the 'Connect' option. Other visible options include 'View details', 'Manage instance state', 'Instance settings', 'Networking', 'Security', 'Image and templates', 'Monitor and troubleshoot', and 'Public IP'. The rest of the interface is the same, displaying the list of instances and the detailed view for instance i-00d90e3ce9c2cf79b.

66.

The screenshot shows the AWS EC2 Instances page with the URL <https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#ConnectToInstance:instanceId=i-00d90e3ce9c2cf79b>. The page title is "Connect to instance". It displays instructions for connecting to the instance i-00d90e3ce9c2cf79b (app1) using an SSH client. It includes a command example: `ssh -i "newkey.pem" ec2-user@10.0.14.126`. A note states: "Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name."

67.

The screenshot shows the AWS EC2 Instances page with the URL <https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#Instances:instanceId=i-00d90e3ce9c2cf79b>. The page title is "Instance ID". It displays the instance ID i-00d90e3ce9c2cf79b (app1) and provides the same set of connection instructions as the previous screenshot, including the command `ssh -i "newkey.pem" ec2-user@10.0.14.126` and the note about the AMI user name.

68.

The screenshot shows a CloudShell window with the AWS logo at the top. The terminal window displays a login banner for Amazon Linux 2023, followed by a ping command to google.com, and a series of commands related to generating and saving a key.pem file. The session ends with a note about Google's ping statistics and a chmod command.

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec  3 10:01:33 2023 from 3.16.146.3
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.250.190.110) 56(84) bytes of data.
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=1 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=2 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=3 ttl=108 time=16.7 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 16.656/16.696/16.725/0.029 ms
[ec2-user@ip-10-0-73-23 ~]$ sudo nano key.pem
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ rm key.pem.save
rm: remove write-protected regular file 'key.pem.save'?
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem

i-017cb989e00cf53f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23
```

69.

The screenshot shows a CloudShell window with the AWS logo at the top. It displays the instance ID i-00d90e3ce9c2cf79b and its name (app1). Below the instance details, there is a numbered list of steps for connecting to the instance via SSH. Step 4 includes a command to copy, which is highlighted with a green checkmark and the text "Command copied". A note at the bottom provides a tip about user names.

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is newkey.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 newkey.pem
4. Connect to your instance using its Private IP:
10.0.14.126

ssh -i "newkey.pem" ec2-user@10.0.14.126

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check the AMI owner has changed the default AMI user name.

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec  3 10:01:33 2023 from 3.16.146.3
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.250.190.110) 56(84) bytes of data.
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=1 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=2 ttl=108 time=16.7 ms
64 bytes from ord37s35-in-f14.1e100.net (142.250.190.110): icmp_seq=3 ttl=108 time=16.7 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 16.656/16.696/16.725/0.029 ms
[ec2-user@ip-10-0-73-23 ~]$ sudo nano key.pem
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ rm key.pem.save
rm: remove write-protected regular file 'key.pem.save'?
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126

i-017cb989e00cf53f4 (web)

PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23
```

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 16.656/16.696/16.725/0.029 ms
[ec2-user@ip-10-0-73-23 ~]$ sudo nano key.pem
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ rm key.pem.save
rm: remove write-protected regular file 'key.pem.save'?
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126
The authenticity of host '10.0.14.126 (10.0.14.126)' can't be established.
ED25519 key fingerprint is SHA256:PoefIpbKuxie9o1/xgv/BSAnEkdObc3ZG6dDSZOpTCjro.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.14.126' (ED25519) to the list of known hosts.

[ec2-user@ip-10-0-14-126 ~]$
```

```

[ec2-user@ip-10-0-14-126 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126
The authenticity of host '10.0.14.126 (10.0.14.126)' can't be established.
ED25519 key fingerprint is SHA256:PoefHpbKux+e9o1/xqv/RSAnEd0bc3ZG6dsZOpTCjro.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.14.126' (ED25519) to the list of known hosts.

,
  _#
~\_ ##### Amazon Linux 2023
~~ \#####\
~~ \###|
~~ \#/ __ https://aws.amazon.com/linux/amazon-linux-2023
~~ V-' '->
~~ / \
~~ .-' _/ /
~~ /_/
~~ /m/

[ec2-user@ip-10-0-14-126 ~]$ ping google.com
PING google.com (142.250.191.238) 56(84) bytes of data.

```

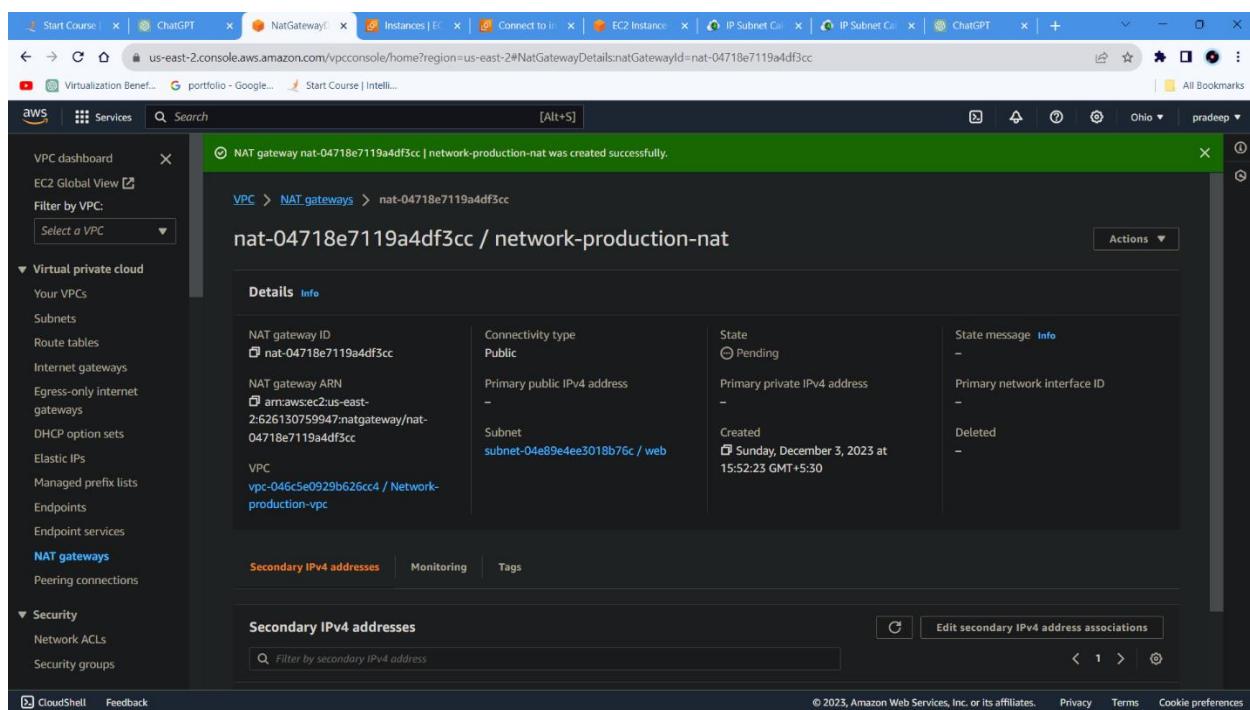
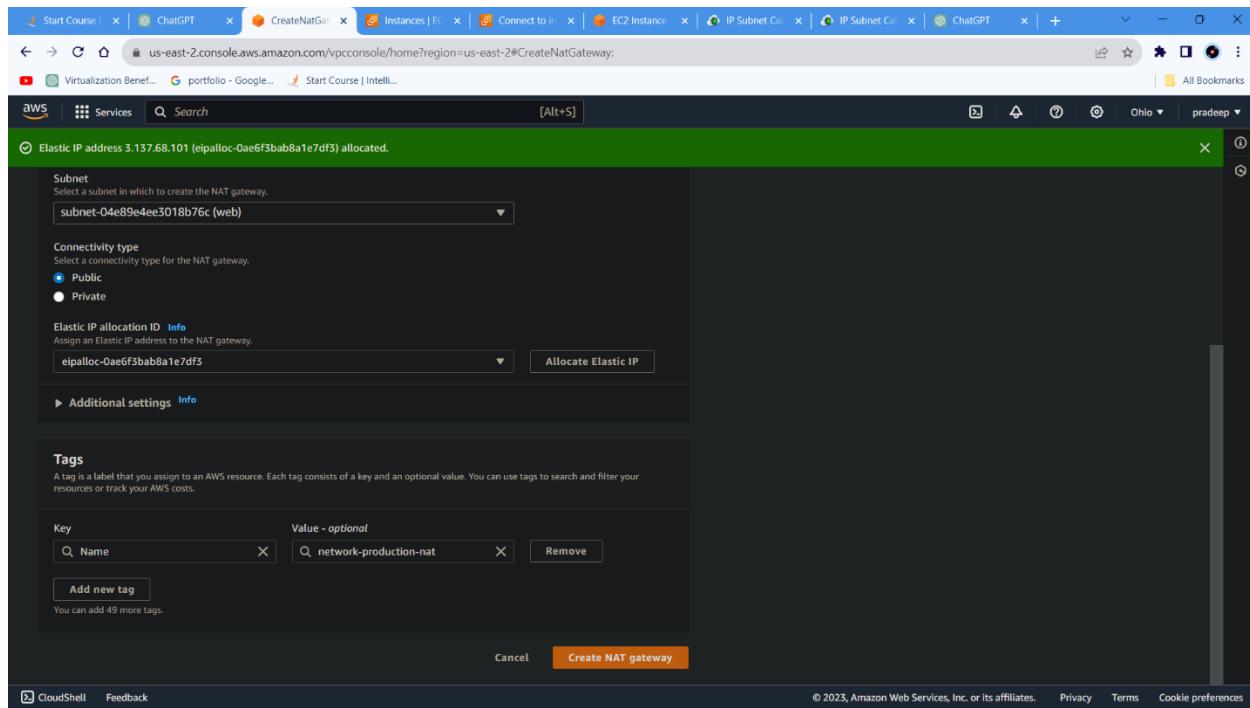
i-017cb989e00cf53f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23

Name	NAT gateway ID	Connectivity ...	State	State message	Primary public IP...	Primary privat

Select a NAT gateway

The screenshot shows the 'Create NAT gateway' configuration page in the AWS VPC console. The 'Name - optional' field contains 'network-production-nat'. The 'Subnet' dropdown is set to 'Select a subnet'. Under 'Connectivity type', 'Public' is selected. In the 'Elastic IP allocation ID' section, 'Select an Elastic IP' is chosen, and the 'Allocate Elastic IP' button is visible. A link to 'Additional settings' is present. The bottom of the page includes standard AWS navigation links like CloudShell and Feedback.

This screenshot shows the same configuration page after an Elastic IP has been allocated. A green banner at the top states 'Elastic IP address 3.157.68.101 (eipalloc-0ae6f5bab8a1e7df5) allocated.' The rest of the interface is identical to the previous screenshot, including the 'Name' field, 'Subnet' selection, connectivity type, and EIP allocation options.



The screenshot shows the AWS VPC Route Tables page. On the left, there's a navigation sidebar with sections like VPC dashboard, EC2 Global View, Filter by VPC, Virtual private cloud (Your VPCs, Subnets, Route tables), Security (Network ACLs, Security groups), and CloudShell/Feedback. The main content area has a title "Route tables (4) Info" and a search bar. A table lists four route tables:

Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC
-	rtb-0a44f83d9f1b19a2d	-	-	Yes	vpc-0f4523b2ab35d0f08
-	rtb-09d3f5523e28461	-	-	Yes	vpc-0a6057a365f2533c81
network-production-RT	rtb-046ac920c29cd79ee	subnet-04e89e4ee3018b...	-	No	vpc-046c5e0929b626cc41
-	rtb-072954dk95ffa65cc	-	-	Yes	vpc-046c5e0929b626cc41

Below the table, there's a section titled "Select a route table".

The screenshot shows the "Create route table" wizard. The first step, "Route table settings", is displayed. It includes fields for "Name - optional" (Nat-RT), "VPC" (vpc-046c5e0929b626cc4 (Network-production-vpc)), and "Tags". Under "Tags", there's a table with one entry: "Key" (Q Name) and "Value - optional" (Q Nat-RT). There are "Add new tag" and "Remove" buttons. At the bottom are "Cancel" and "Create route table" buttons.

The screenshot shows the AWS VPC Route Tables page. A green success message at the top states: "Route table rtb-0f8d16e7043daf9fc | Nat-RT was created successfully." The main content area displays the details for the newly created route table "rtb-0f8d16e7043daf9fc / Nat-RT". The "Details" tab is selected, showing the following information:

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-0f8d16e7043daf9fc	No	-	-
VPC	Owner ID		
vpc-046c5e0929b626cc4 Network-production-vpc	626130759947		

Below the details, there are tabs for "Routes", "Subnet associations", "Edge associations", "Route propagation", and "Tags". The "Routes" tab is selected, showing one route entry:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

At the bottom right of the main content area, there are "Both" and "Edit routes" buttons. The left sidebar shows the navigation path: VPC > Route tables > rtb-0f8d16e7043daf9fc. The "Route tables" section is expanded, showing other route tables like "network-production-RT" and "Nat-RT".

The screenshot shows the AWS VPC Route Tables list page. The main content area displays a table titled "Route tables (5)" with the following data:

Name	Route table ID	Explicit subnet associati...	Edge associations	Main	VPC
rtb-0a44f83d9f1b19a2d	-	-	-	Yes	vpc-0f4523b2ab35df0f8
rtb-09d3f5523e268461	-	-	-	Yes	vpc-0e6057a365f2533e8
network-production-RT	rtb-046ac920c29cd79ee	subnet-04e89e4ee3018b...	-	No	vpc-046c5e0929b626cc4
Nat-RT	rtb-0f8d16e7043daf9fc	-	-	Yes	vpc-046c5e0929b626cc4

Below the table, there is a section titled "Select a route table" with three buttons: "Create route table", "Actions", and a search bar.

Screenshot of the AWS VPC Route Tables page:

Route tables (1/5) Info

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
-	rtb-0a44f83d9f1b19a2d	-	-	Yes	vpc-0f4523b2ab35df0f8
-	rtb-09d33f5525e268461	-	-	Yes	vpc-0a6057a365f2533c8
network-production-RT	rtb-046a6920c29cd79ee	subnet-04e89e4ee3018b...	-	No	vpc-046c5e0929b626cc4
-	rtb-072954dc95ffa65cc	-	-	Yes	vpc-046c5e0929b626cc4
Nat-RT	rtb-0f8d16e7043daf9fc	-	-	No	vpc-046c5e0929b626cc4

rtb-0f8d16e7043daf9fc / Nat-RT

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

Routes (1)

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Screenshot of the AWS VPC Edit Routes page:

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
Q. 0.0.0.0/0	NAT Gateway	-	No
	nat-04718e7119a4df3cc		

Add route

Buttons: Cancel, Preview, Save changes

The screenshot shows the AWS VPC Route Tables page. A prominent green success message at the top states: "Updated routes for rtb-0f8d16e7043daf9fc / Nat-RT successfully". Below this, the route table details are shown: Route table ID (rtb-0f8d16e7043daf9fc), Main (No), VPC (vpc-046c5e0929b626cc4 | Network-production-vpc), Owner ID (626130759947). The "Routes" tab is selected, displaying two routes:

Destination	Target	Status	Propagated
0.0.0.0/0	nat-04718e7119a4df3cc	Active	No
10.0.0.0/16	local	Active	No

This screenshot is identical to the one above, but the "Actions" dropdown menu is open on the right side of the page. The visible options are: Set main route table, Edit subnet associations, Edit edge associations, Edit route propagation, Edit routes, Manage tags, and Delete.

Screenshot of the AWS VPC console showing the 'Edit subnet associations' page for a route table.

Available subnets (2/5)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
db	subnet-01814985b7e28d2cc	10.0.48.0/20	-	Main (rtb-072954dc95ffa65cc)
db-chache	subnet-0a6d37403fd8d39e5	10.0.32.0/20	-	Main (rtb-072954dc95ffa65cc)
app1	subnet-0b13aa0d4eb8c74a9	10.0.0.0/20	-	Main (rtb-072954dc95ffa65cc)
web	subnet-04e89e4ee3018b76c	10.0.64.0/20	-	rtb-046ac920c29cd79ee / network-pro...
app2	subnet-06f00917996347439	10.0.16.0/20	-	Main (rtb-072954dc95ffa65cc)

Selected subnets

- subnet-0b13aa0d4eb8c74a9 / app1
- subnet-0a6d37403fd8d39e5 / db-chache

Actions: Cancel, Save associations

Screenshot of the AWS VPC console showing the 'RouteTableDetails' page for a specific route table.

Updated routes for rtb-0f8d16e7043daf9fc / Nat-RT successfully

You have successfully updated subnet associations for rtb-0f8d16e7043daf9fc / Nat-RT.

rtb-0f8d16e7043daf9fc / Nat-RT

Details **Info**

Route table ID rtb-0f8d16e7043daf9fc	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-046c5e0929b626cc4 Network-production-vpc	Owner ID 626130759947		

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	nat-04718e7119a4df3cc	Active	No
10.0.0.0/16	local	Active	No

Actions: Both, Edit routes

Screenshot of AWS CloudShell showing terminal session on an EC2 instance.

```

[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ rm key.pem.save
rm: remove write-protected regular file `key.pem.save'? [y/N] y
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126
The authenticity of host '10.0.14.126 (10.0.14.126)' can't be established.
ED25519 key fingerprint is SHA256:PoeFhpKux+e9o1/xgv/RSAAnEd0bc3ZG6dsZopTCjro.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.14.126' (ED25519) to the list of known hosts.

[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.250.191.238) 56(84) bytes of data.
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=534 ttl=52 time=17.1 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=535 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=536 ttl=52 time=16.3 ms

[ec2-user@ip-10-0-14-126 ~]$ ping google.com
PING google.com (142.250.191.238) 56(84) bytes of data.
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=534 ttl=52 time=17.1 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=535 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=536 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=537 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=538 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=539 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=540 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=541 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=542 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=543 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=544 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=545 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=546 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=547 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=548 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=549 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=550 ttl=52 time=16.3 ms

[ec2-user@ip-10-0-14-126 ~]$ i-017cb989e00cf53f4 (web)
Public IPs: 3.141.152.51 Private IPs: 10.0.73.23

```

Screenshot of AWS CloudShell showing terminal session on an EC2 instance.

```

[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ rm key.pem.save
rm: remove write-protected regular file `key.pem.save'? [y/N] y
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126
The authenticity of host '10.0.14.126 (10.0.14.126)' can't be established.
ED25519 key fingerprint is SHA256:PoeFhpKux+e9o1/xgv/RSAAnEd0bc3ZG6dsZopTCjro.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.14.126' (ED25519) to the list of known hosts.

[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.250.191.238) 56(84) bytes of data.
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=534 ttl=52 time=17.1 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=535 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=536 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=537 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=538 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=539 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=540 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=541 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=542 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=543 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=544 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=545 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=546 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=547 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=548 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=549 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=550 ttl=52 time=16.3 ms

[ec2-user@ip-10-0-14-126 ~]$ ping google.com
PING google.com (142.250.191.238) 56(84) bytes of data.
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=534 ttl=52 time=17.1 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=535 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=536 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=537 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=538 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=539 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=540 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=541 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=542 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=543 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=544 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=545 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=546 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=547 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=548 ttl=52 time=16.3 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=549 ttl=52 time=16.4 ms
64 bytes from ord38s32-in-f14.le100.net (142.250.191.238): icmp_seq=550 ttl=52 time=16.3 ms

[ec2-user@ip-10-0-14-126 ~]$ i-017cb989e00cf53f4 (web)
Public IPs: 3.141.152.51 Private IPs: 10.0.73.23

```

The screenshot shows the AWS CloudShell interface. A terminal window titled "Windows PowerShell" is open, displaying the command: `PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51`. The terminal window has a sidebar with "EC2 Instance Connect" options and a "Command" section containing the command. The main pane is black, indicating the command is still executing.

The screenshot shows a web browser window for the Intellipaat platform (<https://lms.intellipaat.com/start-course/>). On the left, there is a navigation sidebar with "SELF-PACED" and "LIVE CLASSES" sections. The main content area shows a "Windows PowerShell" window with the following output:

```
PS C:\Users\pradeep acharya> cd downloads
PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
The authenticity of host '3.141.152.51' (3.141.152.51) can't be established.
ED25519 key fingerprint is SHA256:RhvWt+ke02i2jNmdcQpawDRag1kOumkBMEHaVdIV7g.
This key is not known by any other name
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.141.152.51' (ED25519) to the list of known hosts.

#
```

The right side of the screen shows a vertical sidebar with course-related icons like "aws-intellipaat", "awsDB", and "Kinesis".

```

PS C:\Users\pradeep acharya> cd downloads
PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
The authenticity of host '3.141.152.51 (3.141.152.51)' can't be established.
ED25519 key fingerprint is SHA256:RhVwI+ze02i2jNmdcQpawDRag1kOumkBMEHaVdIV7g.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.141.152.51' (ED25519) to the list of known hosts.

# 
#_###_ Amazon Linux 2023
##_\####\
##_#\#
##_#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
##_#/_>
##_/_/_
/_m_/

Last login: Sun Dec 3 10:07:08 2023 from 3.16.146.5
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ 

```

```

-bash: ping: command not found
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.251.32.14) 56(84) bytes of data.
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=1 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=2 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=3 ttl=109 time=16.8 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=4 ttl=109 time=16.7 ms
```
-- google.com ping statistics --
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 16.688/16.722/16.763/0.828 ms
[ec2-user@ip-10-0-73-23 ~]$ exit
logout
Connection to 3.141.152.51 closed.
PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51

#_###_ Amazon Linux 2023
##_\####\
##_#\#
##_#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
##_#/_>
##_/_/_
/_m_/

Last login: Sun Dec 3 10:32:51 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$

```

EC2 Dashboard

Instances (1/6) **Info**

| Name        | Instance ID          | Instance state | Instance type | Status check      | Alarm     |
|-------------|----------------------|----------------|---------------|-------------------|-----------|
| web         | i-017cb989e00cf53f4  | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| db-cache    | i-0e6afe19add716e    | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| <b>app1</b> | i-00d90e3ce9c2cf79b  | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| db          | i-0ccb86cbc1fa144    | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| no          | i-0aa7cf3d1fb8ceef   | Terminated     | t2.micro      | -                 | No alarm  |
| app2        | i-0bf1f0945a7af737b5 | Running        | t2.micro      | 2/2 checks passed | No alarms |

Instance: i-00d90e3ce9c2cf79b (app1)

Details Security Networking Storage Status checks Monitoring Tags

Instance summary

Instance ID: i-00d90e3ce9c2cf79b (app1)

Public IPv4 address: 10.0.14.126

Private IPv4 addresses: 10.0.14.126

Instance state: Running

Public IP4 DNS: ip-10-0-14-126.us-east-2.compute.internal

Hostname type: IP name: ip-10-0-14-126.us-east-2.compute.internal

EC2 > Instances > i-00d90e3ce9c2cf79b > Connect to instance

Connect t

Connect to your instance

EC2 Instance Co

Instance ID: i-00d90e3ce9c2cf79b

1. Open your browser  
2. Locate your instance  
3. Run this command  
4. Connect to your instance

ssh -i "newkey.pem" ec2-user@10.0.14.126

Note: In the AMI configuration, select "Generate a new key pair".

```

-bash: ping: command not found
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.251.32.14) 56(84) bytes of data.
64 bytes from ord38s33-in-f14.1e180.net (142.251.32.14): icmp_seq=1 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e180.net (142.251.32.14): icmp_seq=2 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e180.net (142.251.32.14): icmp_seq=3 ttl=109 time=16.8 ms
64 bytes from ord38s33-in-f14.1e180.net (142.251.32.14): icmp_seq=4 ttl=109 time=16.7 ms
...
google.com ping statistics:
4 packets transmitted, 0 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 16.688/16.722/16.763/0.028 ms
[ec2-user@ip-10-0-73-23 ~]$ exit
Logout
Connection to 3.141.152.51 closed.
PS C:\Users\pradeep.acharya\Downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Sun Dec 3 10:32:51 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "key.pem" ec2-user@10.0.14.126

```

Start Course | In | ChatGPT | RouteTables | VPC | Connect to instance | Connect to instance | IP Subnet Calc... | IP Subnet Calc... | ChatGPT | +

aws Services Search [Alt+S]

EC2 > Instances > i-00d90e3ce9c2cf79b > Connect to instance

**Connect to** ec2-user@ip-10-0-73-23~

```
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.251.32.14) 56(84) bytes of data.
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=1 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=2 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=3 ttl=109 time=16.8 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=4 ttl=109 time=16.7 ms

```

EC2 Instance Configuration

Instance ID: i-00d90e3ce9c2cf79b

1. Open an S 2. Locate you... 3. Run this co... 4. Connect to... 10.0.0.1

Command Line: ssh -i "newkey.pem" ec2-user@ip-10-0-73-23

Note: In the AMI configuration, the key pair is set to "newkey.pem".

CloudShell Feedback

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Start Course | In | ChatGPT | RouteTables | VPC | Connect to instance | Connect to instance | IP Subnet Calc... | IP Subnet Calc... | ChatGPT | +

aws Services Search [Alt+S]

EC2 > Instances > i-00d90e3ce9c2cf79b > Connect to instance

**Connect to** ec2-user@ip-10-0-73-23~

```
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.251.32.14) 56(84) bytes of data.
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=1 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=2 ttl=109 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=3 ttl=109 time=16.8 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=4 ttl=109 time=16.7 ms

```

EC2 Instance Configuration

Instance ID: i-00d90e3ce9c2cf79b

1. Open an S 2. Locate you... 3. Run this co... 4. Connect to... 10.0.0.1

Command Line: ssh -i "newkey.pem" ec2-user@ip-10-0-73-23

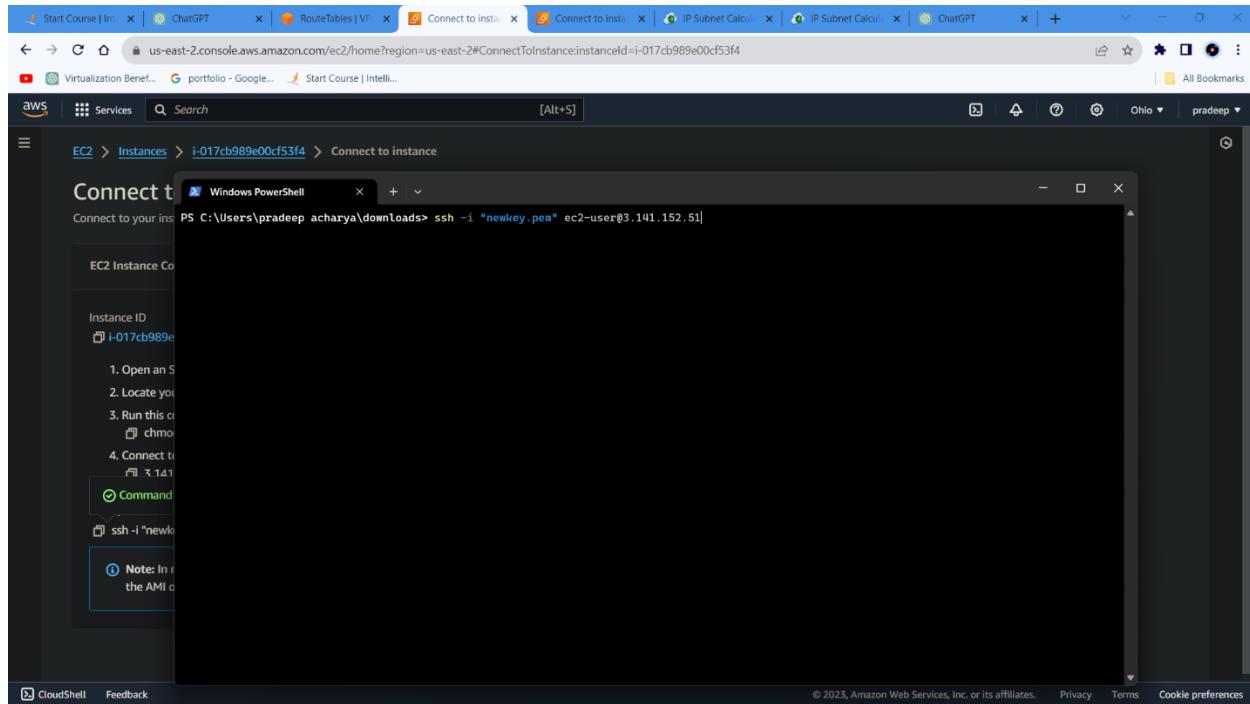
Note: In the AMI configuration, the key pair is set to "newkey.pem".

CloudShell Feedback

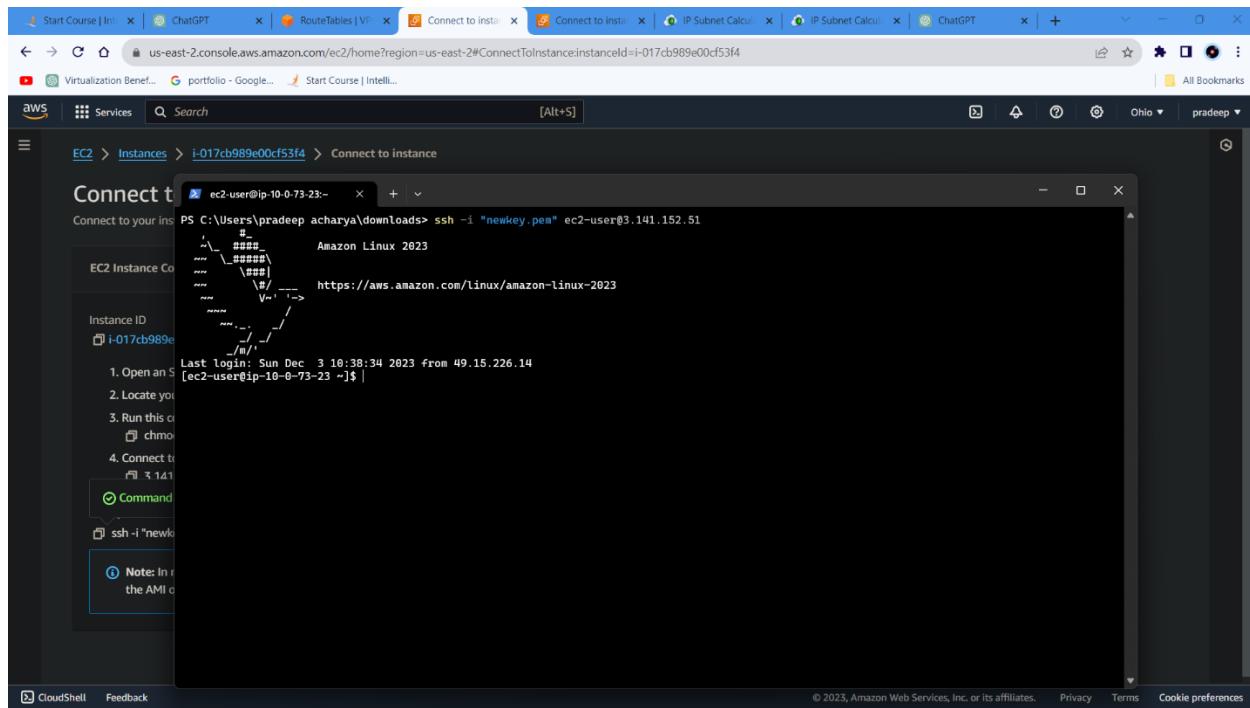
© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (142.251.32.14) 56(84) bytes of data.
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=1 ttl=189 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=2 ttl=189 time=16.7 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=3 ttl=189 time=16.8 ms
64 bytes from ord38s33-in-f14.1e100.net (142.251.32.14): icmp_seq=4 ttl=189 time=16.7 ms
*c
-- google.com ping statistics --
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 16.688/16.722/16.763/0.028 ms
[ec2-user@ip-10-0-73-23 ~]$ exit
logout
Connection to 3.141.152.51 closed.
PS C:\Users\pradeep.acharya\Downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
.
.
.
Amazon Linux 2023
.
.
.
https://aws.amazon.com/linux/amazon-linux-2023
.
.
.
Last login: Sun Dec 3 10:32:51 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "key.pem" ec2-user@10.0.14.126
The authenticity of host '10.0.14.126 (10.0.14.126)' can't be established.
ED25519 key fingerprint is SHA256:PoeffhpKux+e9o1/xgv/RSAAnEdObc3ZG6dSZOpTCjro.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.14.126' (ED25519) to the list of known hosts.
Load key "key.pem": Permission denied
ec2-user@10.0.14.126: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
```

```
*c
-- google.com ping statistics --
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 16.688/16.722/16.763/0.028 ms
[ec2-user@ip-10-0-73-23 ~]$ exit
logout
Connection to 3.141.152.51 closed.
PS C:\Users\pradeep.acharya\Downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
.
.
.
Amazon Linux 2023
.
.
.
https://aws.amazon.com/linux/amazon-linux-2023
.
.
.
Last login: Sun Dec 3 10:32:51 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "key.pem" ec2-user@10.0.14.126
The authenticity of host '10.0.14.126 (10.0.14.126)' can't be established.
ED25519 key fingerprint is SHA256:PoeffhpKux+e9o1/xgv/RSAAnEdObc3ZG6dSZOpTCjro.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.14.126' (ED25519) to the list of known hosts.
Load key "key.pem": Permission denied
ec2-user@10.0.14.126: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-73-23 ~]$ ping google.com
PING google.com (172.217.1.110) 56(84) bytes of data.
64 bytes from yyz08s09-in-f110.1e100.net (172.217.1.110): icmp_seq=1 ttl=117 time=16.3 ms
64 bytes from yyz08s09-in-f14.1e100.net (172.217.1.110): icmp_seq=2 ttl=117 time=16.3 ms
64 bytes from ord37s51-in-f14.1e100.net (172.217.1.110): icmp_seq=3 ttl=117 time=16.3 ms
64 bytes from mia09s17-in-f14.1e100.net (172.217.1.110): icmp_seq=4 ttl=117 time=16.3 ms
```



A screenshot of the AWS CloudShell interface. The title bar shows the URL `us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#ConnectToInstance:instanceId=i-017cb989e00cf53f4`. The main window is titled "Connect to instance". It displays a Windows PowerShell window with the command `PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51`. To the left of the PowerShell window is a sidebar with the heading "EC2 Instance Connect" and a list of steps: 1. Open an SSM session, 2. Locate your instance, 3. Run this command, 4. Connect to the instance. Step 3 is highlighted with a blue border. Below the steps is a "Command" section containing the SSH command. A note at the bottom says "Note: Instances must be running to connect via the AMI connection." At the bottom of the sidebar are "CloudShell" and "Feedback" buttons.



A screenshot of the AWS CloudShell interface, identical to the first one but showing the result of the command execution. The PowerShell window now displays the output of the SSH session. It shows the Amazon Linux 2023 logo, the URL `https://aws.amazon.com/linux/amazon-linux-2023`, and the message "Last login: Sun Dec 3 10:38:34 2023 from 49.15.226.14 [ec2-user@ip-10-0-73-23 ~]\$". The rest of the interface, including the sidebar and footer, remains the same.

**Instances (1/6) Info**

| Name        | Instance ID          | Instance state | Instance type | Status check      | Alarm     |
|-------------|----------------------|----------------|---------------|-------------------|-----------|
| web         | i-017cb98e00cf53f4   | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| db-chache   | i-0e6afce19add716e   | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| <b>app1</b> | i-00d90e3ce9c2cf79b  | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| db          | i-0cfb86cbc1fa144    | Running        | t2.micro      | 2/2 checks passed | No alarm  |
| no          | i-0aa7cf5d1fb8ceef   | Terminated     | t2.micro      | -                 | No alarm  |
| app2        | i-0bf1f0945a7af737b5 | Running        | t2.micro      | 2/2 checks passed | No alarms |

**Instance: i-00d90e3ce9c2cf79b (app1)**

**Details** | Security | Networking | Storage | Status checks | Monitoring | Tags

**Instance summary**

|                                                                     |                                                                              |                                       |
|---------------------------------------------------------------------|------------------------------------------------------------------------------|---------------------------------------|
| Instance ID<br>i-00d90e3ce9c2cf79b (app1)                           | Public IPv4 address<br>-                                                     | Private IPv4 addresses<br>10.0.14.126 |
| IPv6 address<br>-                                                   | Instance state<br>Running                                                    | Public IPv4 DNS<br>-                  |
| Hostname type<br>IP name: ip-10-0-14-126.us-east-2.compute.internal | Private IP DNS name (IPv4 only)<br>ip-10-0-14-126.us-east-2.compute.internal | Private IP DNS name (IPv6 only)<br>-  |

**EC2 > Instances > i-00d90e3ce9c2cf79b > Connect to instance**

**Connect to instance**

Connect to your instance i-00d90e3ce9c2cf79b (app1) using any of these options

**SSH client**

Instance ID  
i-00d90e3ce9c2cf79b (app1)

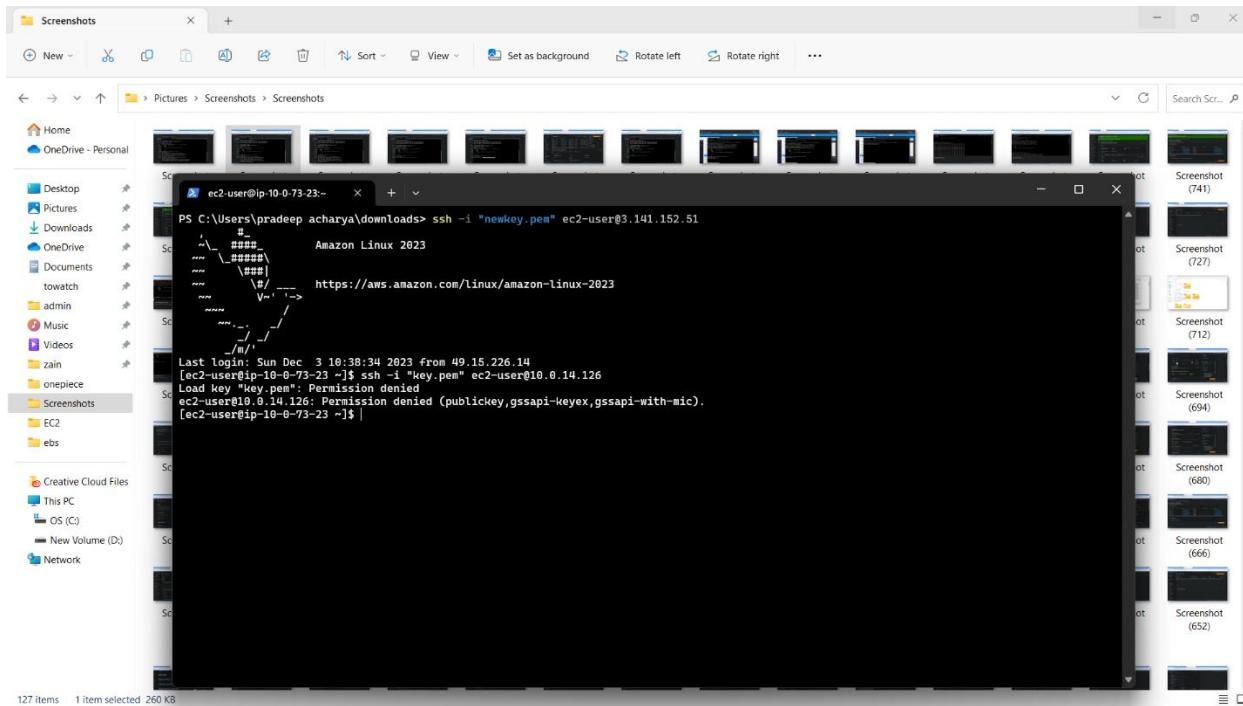
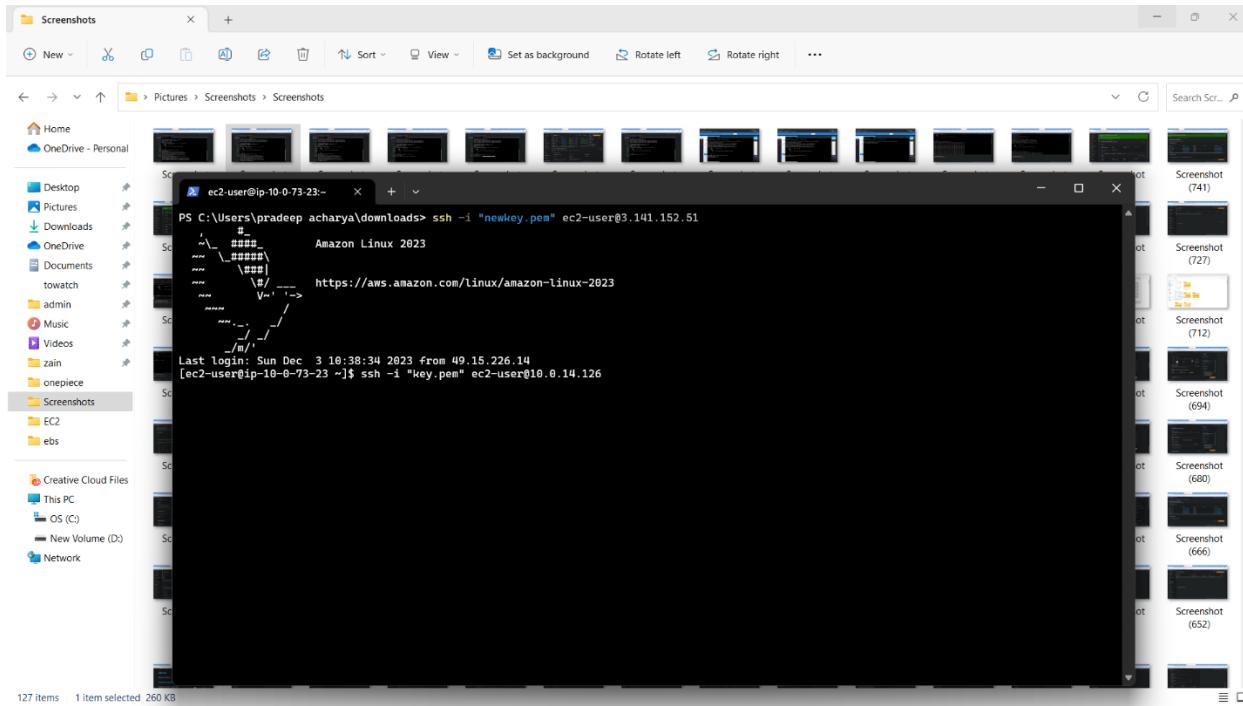
- Open an SSH client.
- Locate your private key file. The key used to launch this instance is newkey.pem
- Run this command, if necessary, to ensure your key is not publicly viewable.  
chmod 400 newkey.pem
- Connect to your instance using its Private IP:  
10.0.14.126

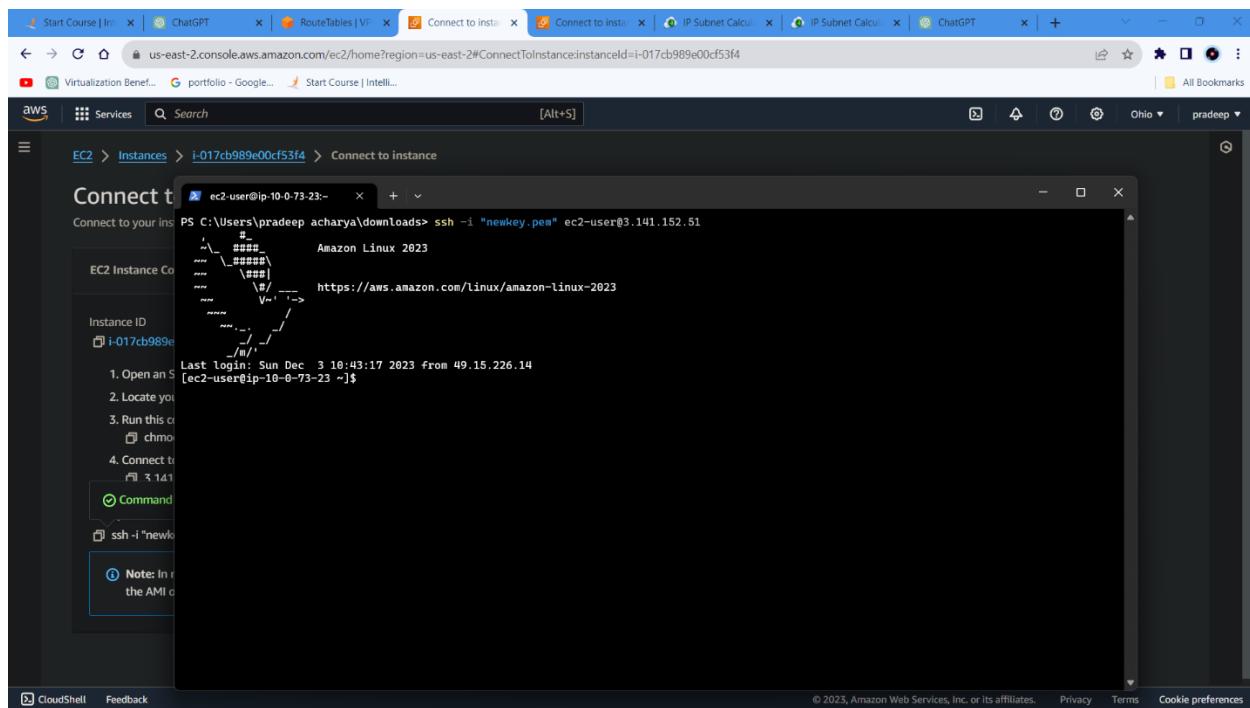
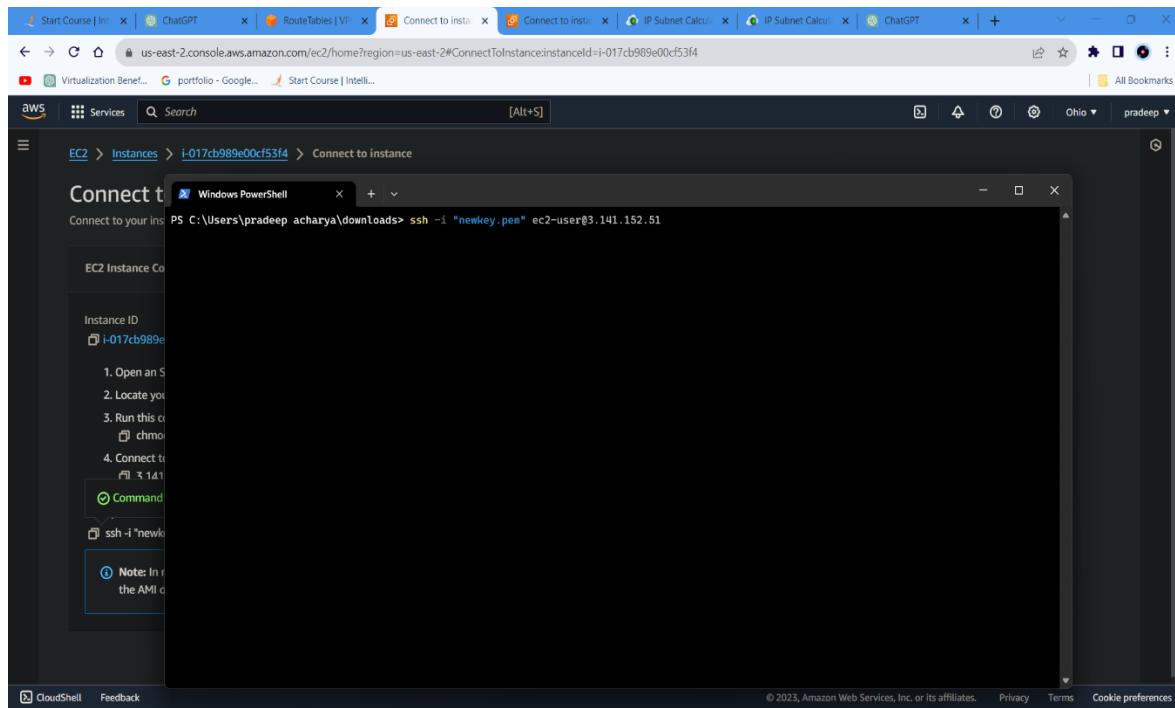
**Command copied**

```
ssh -i "newkey.pem" ec2-user@10.0.14.126
```

**Note:** In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

**Cancel**





The screenshot shows the AWS CloudShell interface. A terminal window titled "Connect to instance" is open, showing an SSH session to an EC2 instance. The command entered is:

```
PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
```

The terminal output shows:

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

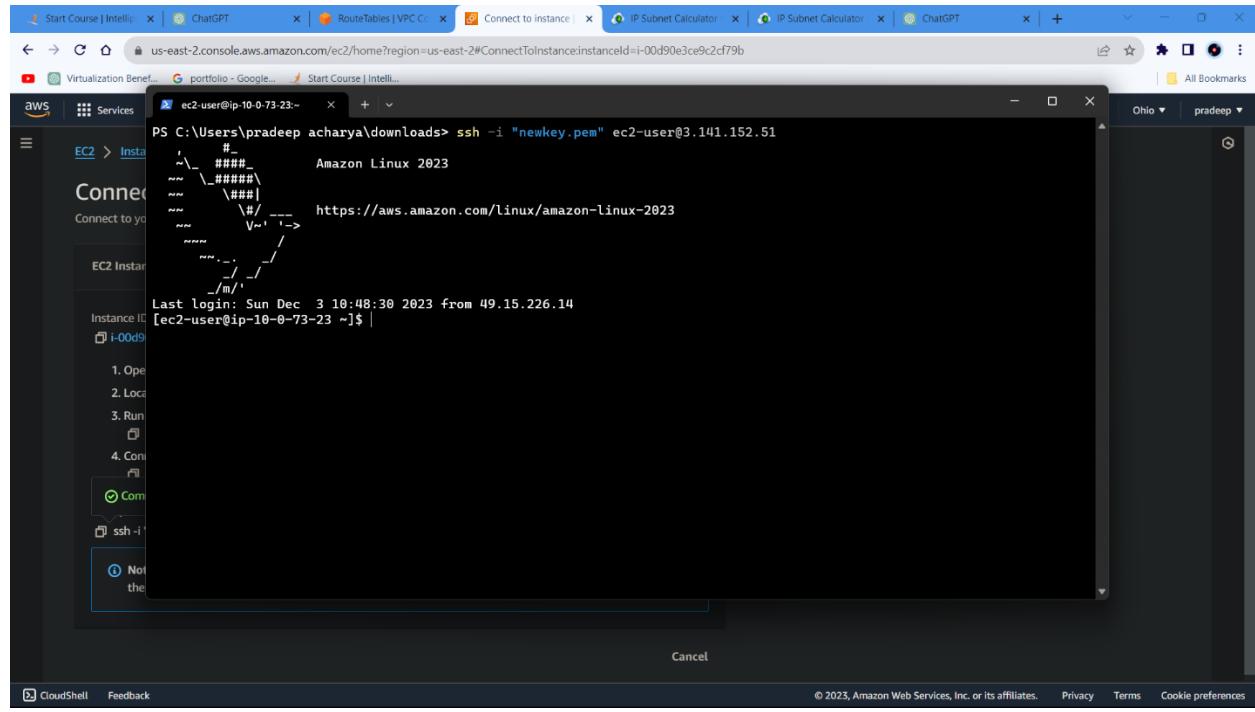
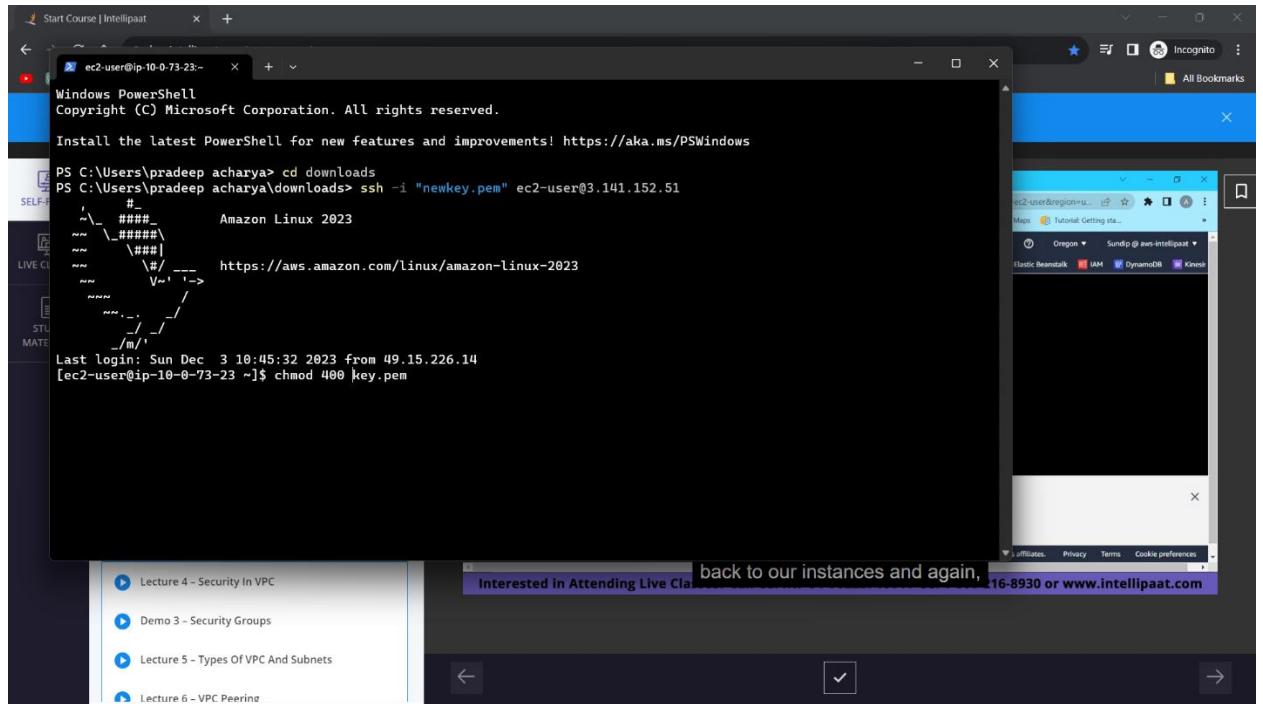
Last login: Sun Dec 3 10:43:17 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "newkey.pem" ec2-user@10.0.14.126
Warning: Identity file newkey.pem not accessible: No such file or directory.
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "key.pem" ec2-user@10.0.14.126
[ec2-user@ip-10-0-73-23 ~]$
```

A note at the bottom of the terminal window says: "Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name."

The screenshot shows the AWS CloudShell interface. A terminal window titled "Windows PowerShell" is open, showing an SSH session to an EC2 instance. The command entered is:

```
PS C:\Users\pradeep acharya> cd downloads
PS C:\Users\pradeep acharya\downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
```

A note at the bottom of the terminal window says: "Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name."



A screenshot of the AWS CloudShell interface. The main window shows a terminal session with the following commands and output:

```
PS C:\Users\pradeep_acharya\Downloads> ssh -i "newkey.pem" ec2-user@3.141.152.51
Last login: Sun Dec 3 10:48:30 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$
```

The sidebar on the left lists EC2 instances, with one named `i-00d9e3ce9c2cf79b` highlighted. The bottom status bar includes links for CloudShell and Feedback, and copyright information for Amazon Web Services.

A screenshot of the AWS CloudShell interface. The terminal session shows the following commands and errors:

```
Last login: Sun Dec 3 10:48:30 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "newkey.pem" ec2-user@10.0.14.126
Warning: Identity file newkey.pem not accessible: No such file or directory.
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "key.pem" ec2-user@10.0.14.126
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126
Load key "key.pem": Permission denied
[ec2-user@ip-10-0-73-23 ~]$
```

The sidebar on the left lists EC2 instances, with one named `i-00d9e3ce9c2cf79b` highlighted. The bottom status bar includes links for CloudShell and Feedback, and copyright information for Amazon Web Services.

The screenshot shows a CloudShell session titled "ec2-user@ip-10-0-14-126~". The terminal window displays the following command-line session:

```
Last login: Sun Dec 3 10:48:30 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "newkey.pem" ec2-user@10.0.14.126
Warning: Identity file newkey.pem not accessible: No such file or directory.
[ec2-user@10.0.14.126: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-73-23 ~]$ ssh -i "key.pem" ec2-user@10.0.14.126
Load key "key.pem": Permission denied
[ec2-user@10.0.14.126: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.14.126
```

After this, the user runs a ping command:

```
Last login: Sun Dec 3 10:15:40 2023 from 10.0.73.23
[ec2-user@ip-10-0-14-126 ~]$ ping google.com
PING google.com (172.217.5.14) 56(84) bytes of data.
64 bytes from ord38s19-in-f14.1e100.net (172.217.5.14): icmp_seq=1 ttl=116 time=16.0 ms
64 bytes from lga15s49-in-f14.1e100.net (172.217.5.14): icmp_seq=2 ttl=116 time=16.3 ms
64 bytes from ord38s19-in-f14.1e100.net (172.217.5.14): icmp_seq=3 ttl=116 time=16.2 ms
64 bytes from lga15s49-in-f14.1e100.net (172.217.5.14): icmp_seq=4 ttl=116 time=16.2 ms
```

Then, the user runs a ping statistics command:

```
-- google.com ping statistics --
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 16.166/16.306/16.593/0.172 ms
the [ec2-user@ip-10-0-14-126 ~]$
```

The screenshot shows the "Subnet settings" configuration page for a VPC subnet. The page title is "Subnet 1 of 1".

**Subnet name:** web-2

**Availability Zone:** No preference

**IPv4 VPC CIDR block:** 10.1.0.0/16

**IPv4 subnet CIDR block:** 10.1.0.0/20 (4,096 IPs)

**Tags - optional:**

| Key  | Value - optional |
|------|------------------|
| Name | web-2            |

You can add 49 more tags.

Screenshot of the AWS VPC Subnets page showing two newly created subnets:

| Name  | Subnet ID                | State     | VPC                             | IPv4 CIDR    |
|-------|--------------------------|-----------|---------------------------------|--------------|
| db-2  | subnet-0c7d08474573014c2 | Available | vpc-0a6057a365f2533c8   deve... | 10.1.16.0/20 |
| web-2 | subnet-0a77f803c3857832c | Available | vpc-0a6057a365f2533c8   deve... | 10.1.0.0/20  |

Screenshot of the AWS VPC Subnets page showing the same two subnets, with the "web-2" subnet selected:

| Name         | Subnet ID                       | State            | VPC                                    | IPv4 CIDR          |
|--------------|---------------------------------|------------------|----------------------------------------|--------------------|
| app1         | subnet-0a77f803c3857832c/4a9    | Available        | vpc-046c5e0929b626cc4   Net...         | 10.0.0.0/20        |
| web          | subnet-04e89e4ee3018b76c        | Available        | vpc-046c5e0929b626cc4   Net...         | 10.0.64.0/20       |
| -            | subnet-0631dfa5d697eec3         | Available        | vpc-0f4523b2ab35df0f8                  | 172.31.0.0/20      |
| app2         | subnet-0f00917996347439         | Available        | vpc-046c5e0929b626cc4   Net...         | 10.0.16.0/20       |
| db-2         | subnet-0c7d08474573014c2        | Available        | vpc-0a6057a365f2533c8   deve...        | 10.1.16.0/20       |
| <b>web-2</b> | <b>subnet-0a77f803c3857832c</b> | <b>Available</b> | <b>vpc-0a6057a365f2533c8   deve...</b> | <b>10.1.0.0/20</b> |

Details for the selected subnet:

| Details                               | Flow logs                                                                       | Route table               | Network ACL                     | CIDR reservations | Sharing | Tags |
|---------------------------------------|---------------------------------------------------------------------------------|---------------------------|---------------------------------|-------------------|---------|------|
| <b>Details</b>                        |                                                                                 |                           |                                 |                   |         |      |
| Subnet ID<br>subnet-0a77f803c3857832c | Subnet ARN<br>arn:aws:ec2:us-east-2:62613075947:subnet/subnet-0a77f803c3857832c | State<br><b>Available</b> | IPv4 CIDR<br><b>10.1.0.0/20</b> |                   |         |      |
| Available IPv4 addresses              |                                                                                 |                           | Availability Zone               |                   |         |      |
|                                       |                                                                                 |                           | Availability Zone ID            |                   |         |      |

Screenshot of the AWS VPC Subnets page showing two subnets created:

| Name         | Subnet ID                       | State            | VPC                                |
|--------------|---------------------------------|------------------|------------------------------------|
| app-1        | subnet-0a77f803c3857832c        | Available        | vpc-046c5e0929b626cc4   Net        |
| web          | subnet-04e89e4ee3018b76c        | Available        | vpc-046c5e0929b626cc4   Net        |
| -            | subnet-0631dfaf5d697eec3        | Available        | vpc-0f4523b2ab35df0f8              |
| app2         | subnet-0f00917996347439         | Available        | vpc-046c5e0929b626cc4   Net        |
| db-2         | subnet-0c7d08474573014c2        | Available        | vpc-0a6057a365f2533c8   dev        |
| <b>web-2</b> | <b>subnet-0a77f803c3857832c</b> | <b>Available</b> | <b>vpc-0a6057a365f2533c8   dev</b> |

Actions menu for the selected subnet (web-2):

- Create flow log
- Edit subnet settings
- Edit IPv6 CIDs
- Edit network ACL association
- Edit route table association
- Edit CIDR reservations
- Share subnet
- Manage tags
- Delete subnet

Screenshot of the Edit Subnet Settings page for subnet-0a77f803c3857832c:

### Edit subnet settings

**Subnet**

|                          |       |
|--------------------------|-------|
| Subnet ID                | Name  |
| subnet-0a77f803c3857832c | web-2 |

**Auto-assign IP settings**

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

- Enable auto-assign public IPv4 address
- Enable auto-assign customer-owned IPv4 address

**Resource-based name (RBN) settings**

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

- Enable resource name DNS A record on launch
- Enable resource name DNS AAAA record on launch

Hostname type:

- Resource name
- IP name

The screenshot shows the 'Auto-assign IP settings' configuration page in the AWS VPC console. It includes sections for enabling auto-assignment of public IPv4 addresses, resource-based name (RBN) settings, and DNS64 settings. The 'Save' button is highlighted.

**Auto-assign IP settings**

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

Enable auto-assign public IPv4 address [Info](#)

Enable auto-assign customer-owned IPv4 address [Info](#)  
Option disabled because no customer owned pools found.

**Resource-based name (RBN) settings** [Info](#)

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

Enable resource name DNS A record on launch [Info](#)

Enable resource name DNS AAAA record on launch [Info](#)

Hostname type [Info](#)

Resource name

IP name

**DNS64 settings**

Enable DNS64 to allow IPv6-only services in Amazon VPC to communicate with IPv4-only services and networks.

Enable DNS64 [Info](#)

Cancel Save

The screenshot shows the 'Internet gateways (2)' list in the AWS VPC console. The table displays two entries: 'igw-05564125e1e40437c' and 'network-production-igw'. The 'Create internet gateway' button is visible at the top right.

| Name                   | Internet gateway ID   | State    | VPC ID                                 | Owner        |
|------------------------|-----------------------|----------|----------------------------------------|--------------|
| -                      | igw-05564125e1e40437c | Attached | vpc-Of4523b2ab35df0f8                  | 626130759947 |
| network-production-igw | igw-0749bb8cf88dbcefa | Attached | vpc-046c5e0929b626cc4   Network-pro... | 626130759947 |

Select an internet gateway above

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Start Course | Info ChatGPT Create internet gateway Instances | EC2 IP Subnet Calculator IP Subnet Calculator ChatGPT New Tab + - X

Virtualization Benefits portfolio - Google... Start Course | Intelli...

aws Services Search [Alt+S]

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="development-igw"/> Remove |
| <a href="#">Add new tag</a>       |                                                     |

You can add 49 more tags.

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

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Start Course | Info ChatGPT igws | VPC Console Instances | EC2 IP Subnet Calculator IP Subnet Calculator ChatGPT New Tab + - X

Virtualization Benefits portfolio - Google... Start Course | Intelli...

aws Services Search [Alt+S]

VPC dashboard X

EC2 Global View

Filter by VPC: [Select a VPC](#)

Virtual private cloud

Your VPCs

Subnets

Route tables

**Internet gateways**

Egress-only internet gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

Peering connections

Security

Network ACLs

Security groups

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

**Internet gateways (1/3) Info**

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

| Name                                                | Internet gateway ID   | State    | VPC ID                                 | Owner        |
|-----------------------------------------------------|-----------------------|----------|----------------------------------------|--------------|
| -                                                   | igw-05564125e1e40437c | Attached | vpc-0f4523b2ab35df0f8                  | 626130759947 |
| network-production-igw                              | igw-0749bb8cf88dbcefa | Attached | vpc-046c5e0929b626cc4   Network-pro... | 626130759947 |
| <input checked="" type="checkbox"/> development-igw | igw-03e919e0abd283506 | Detached | -                                      | 626130759947 |

**igw-03e919e0abd283506 / development-igw**

[Details](#) [Tags](#)

**Details**

|                                                              |                                   |             |                       |
|--------------------------------------------------------------|-----------------------------------|-------------|-----------------------|
| Internet gateway ID<br><a href="#">igw-03e919e0abd283506</a> | State<br><a href="#">Detached</a> | VPC ID<br>- | Owner<br>626130759947 |
|--------------------------------------------------------------|-----------------------------------|-------------|-----------------------|

The screenshot shows the AWS VPC console with the 'Internet gateways' list. The 'development-igw' gateway is selected. Below the table, its detailed information is displayed.

| Name                   | Internet gateway ID          | State           | VPC ID     | Owner        |
|------------------------|------------------------------|-----------------|------------|--------------|
| -                      | igw-05564125e1e40437c        | Attached        | vpc-0f4523 | 626130759947 |
| network-production-igw | igw-0749bb8cf88dbcefa        | Attached        | vpc-046c5e | 626130759947 |
| <b>development-igw</b> | <b>igw-03e919e0abd283506</b> | <b>Detached</b> | -          | 626130759947 |

**igw-03e919e0abd283506 / development-igw**

**Details**

|                                              |                   |             |                       |
|----------------------------------------------|-------------------|-------------|-----------------------|
| Internet gateway ID<br>igw-03e919e0abd283506 | State<br>Detached | VPC ID<br>- | Owner<br>626130759947 |
|----------------------------------------------|-------------------|-------------|-----------------------|

The screenshot shows the 'Attach to VPC' dialog box. It lists available VPCs and allows selecting one via a search bar or AWS CLI command.

**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.

Q vpc-0a6057a365f2533c8

▶ AWS Command Line Interface command

Cancel **Attach internet gateway**

The screenshot shows the AWS VPC Route Tables page. On the left, there's a navigation sidebar with options like VPC dashboard, EC2 Global View, Filter by VPC (with a dropdown menu), Virtual private cloud (Your VPCs, Subnets, Route tables selected), Security (Network ACLs, Security groups), and CloudShell/Feedback. The main content area has a search bar and a table titled "Route tables (5) Info". The table has columns for Name, Route table ID, Explicit subnet associations, Edge associations, Main, and VPC. The data is as follows:

| Name                  | Route table ID        | Explicit subnet associations | Edge associations | Main | VPC                   |
|-----------------------|-----------------------|------------------------------|-------------------|------|-----------------------|
| -                     | rtb-0a44f83d9f1b19a2d | -                            | -                 | Yes  | vpc-0f4523b2ab35df0f8 |
| Nat-RT                | rtb-0f8d16e7043da9fc  | 2 subnets                    | -                 | No   | vpc-046c5e0929b626cc4 |
| -                     | rtb-09d3f5f523e268461 | -                            | -                 | Yes  | vpc-0a6057a365f253c8  |
| network-production-RT | rtb-046ac920c29cd79ee | subnet-04e89e4ee3018b...     | -                 | No   | vpc-046c5e0929b626cc4 |
| -                     | rtb-072954dc95ffa65cc | -                            | -                 | Yes  | vpc-046c5e0929b626cc4 |

Below the table, there's a section titled "Select a route table" with three icons: a magnifying glass, a plus sign, and a minus sign.

This screenshot is identical to the one above, showing the AWS VPC Route Tables page with the same navigation sidebar, search bar, and table of route tables. The data in the table remains the same, listing five route tables with their respective IDs, associations, and VPCs.

**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.  
**development -RT**

VPC  
The VPC to use for this route table.  
**vpc-0a6057a365f2533c8 (development-network-vpc)**

**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>  |
|---------------|--------------------------|
| <b>Q Name</b> | <b>Q development -RT</b> |

**Add new tag**  
You can add 49 more tags.

**Create route table**

**Route table rtb-0a42d02d945304778 | development -RT was created successfully.**

**rtb-0a42d02d945304778 / development -RT**

**Details** Info

|                                                               |                                 |                                   |                        |
|---------------------------------------------------------------|---------------------------------|-----------------------------------|------------------------|
| Route table ID<br><b>rtb-0a42d02d945304778</b>                | Main<br><b>No</b>               | Explicit subnet associations<br>- | Edge associations<br>- |
| VPC<br><b>vpc-0a6057a365f2533c8   development-network-vpc</b> | Owner ID<br><b>626130759947</b> |                                   |                        |

**Routes (1)**

| Destination        | Target       | Status        | Propagated |
|--------------------|--------------|---------------|------------|
| <b>10.1.0.0/16</b> | <b>local</b> | <b>Active</b> | <b>No</b>  |

The screenshot shows the AWS VPC Route Table Details page. A success message at the top states: "Route table rtb-0a42d02d945304778 | development -RT was created successfully." The main content area displays the details of the newly created route table, including its ID (rtb-0a42d02d945304778), VPC (vpc-0a6057a565f2533c8 | development-network-vpc), and owner ID (626130759947). Below this, the "Routes" tab is selected, showing one route entry: Destination 10.1.0.0/16, Target local, Status Active, and Propagated No. The page includes a sidebar with various VPC management options like Internet gateways, Egress-only internet gateways, and Network ACLs.

The screenshot shows the AWS VPC Edit Routes page for the route table rtb-0a42d02d945304778. The "Edit routes" section is active. A table lists existing routes: one for destination 10.1.0.0/16 with target "local" and status "Active". A new route is being added for destination 0.0.0.0/0, with target set to "Internet Gateway" and selected gateway "igw-03e919e0abd283506". The "Add route" button is visible at the bottom left. At the bottom right are "Cancel", "Preview", and "Save changes" buttons. The page footer includes standard AWS links for CloudShell, Feedback, and legal notices.

The screenshot shows the AWS VPC Route Table Details page for route table ID rtb-0a42d02d945304778. A green banner at the top indicates "Updated routes for rtb-0a42d02d945304778 / development -RT successfully". The main section displays the route table's details, including its ID (rtb-0a42d02d945304778), VPC (vpc-06057a365f2533c8), and owner (development-network-vpc). The "Routes" tab is selected, showing two routes:

| Destination | Target                | Status | Propagated |
|-------------|-----------------------|--------|------------|
| 0.0.0.0/0   | igw-03e919e0abd283506 | Active | No         |
| 10.1.0.0/16 | local                 | Active | No         |

The "Actions" menu on the right includes options like Set main route table, Edit subnet associations, Edit edge associations, Edit route propagation, Edit routes, Manage tags, and Delete.

This screenshot is identical to the one above, showing the AWS VPC Route Table Details page for route table ID rtb-0a42d02d945304778. It displays the same green banner, route table details, and two routes in the "Routes" tab. The "Actions" menu is also present on the right side.

The screenshot shows the AWS VPC Route Tables interface. A green banner at the top indicates "Updated routes for rtb-0a42d02d945304778 / development -RT successfully". Below this, the "Edit subnet associations" section is displayed. It shows a table of available subnets:

| Name                                      | Subnet ID                | IPv4 CIDR    | IPv6 CIDR | Route table ID               |
|-------------------------------------------|--------------------------|--------------|-----------|------------------------------|
| db-2                                      | subnet-0c7d08474575014c2 | 10.1.16.0/20 | -         | Main (rtb-09d33f5523e268461) |
| <input checked="" type="checkbox"/> web-2 | subnet-0a77f803c3857832c | 10.1.0.0/20  | -         | Main (rtb-09d33f5523e268461) |

Below the table, a "Selected subnets" section contains the selected subnet: "subnet-0a77f803c3857832c / web-2". At the bottom right are "Cancel" and "Save associations" buttons.

The screenshot shows the AWS EC2 Instances interface. A green banner at the top indicates "Successfully stopped i-0e6afce19add716e, i-00d90e3ce9c2cf79b, i-0b1f0945a7af737b5". Below this, the "Instances (3/5) Info" section is displayed. It shows a table of instances:

| Name                                          | Instance ID         | Instance state | Instance type | Status check      | Alarm status | Availability Zone | Public IPv4 DNS |
|-----------------------------------------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|-----------------|
| <input checked="" type="checkbox"/> db-chache | i-0e6afce19add716e  | Stopped        | t2.micro      | -                 | No alarms    | + us-east-2c      | -               |
| <input checked="" type="checkbox"/> app1      | i-00d90e3ce9c2cf79b | Stopped        | t2.micro      | -                 | No alarms    | + us-east-2c      | -               |
| <input type="checkbox"/> db                   | i-0ccfb86cbcf1fa144 | Running        | t2.micro      | 2/2 checks passed | No alarms    | + us-east-2c      | -               |
| <input type="checkbox"/> web                  | i-017cb989e00cf5f4  | Running        | t2.micro      | 2/2 checks passed | No alarms    | + us-east-2c      | -               |

Below the table, a summary message says "Instances: i-0e6afce19add716e (db-chache), i-00d90e3ce9c2cf79b (app1), i-0b1f0945a7af737b5 (app2)". The "Monitoring" section displays four line charts for CPU utilization, status check failed (any), status check failed (instance), and status check failed (system).

The screenshot shows the AWS EC2 Launch Instances page. In the 'Name and tags' section, the name 'web-2' is entered. Below it, the 'Application and OS Images (Amazon Machine Image)' section is visible, featuring a search bar and a grid of AMI icons for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. A tooltip for the 'Free tier' is displayed over the 'Launch instance' button.

The screenshot shows the AWS EC2 Launch Instances page with the 'Instance type' section selected. It lists the 't2.micro' instance type as 'Free tier eligible'. Below it, the 'Key pair (login)' section shows a dropdown for 'Key pair name - required' with 'newkey' selected, and a 'Create new key pair' button. The 'Network settings' section at the bottom includes a 'Network' dropdown set to 'auto-2023-07-26-00' and an 'Edit' button. A tooltip for the 'Free tier' is also present over the 'Launch instance' button.

| Name      | Instance ID         | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS |
|-----------|---------------------|----------------|---------------|--------------|--------------|-------------------|-----------------|
| db-chache | i-0e6afce19add716e  | Stopped        | t2.micro      | -            | No alarms    | + us-east-2c      | -               |
| app1      | i-00d90e3ce9c2cf79b | Stopped        | t2.micro      | -            | No alarms    | + us-east-2c      | -               |
| db        | i-0cfb86bcf1fa144   | Running        | t2.micro      | -            | No alarms    | + us-east-2c      | -               |
| web-2     | i-08a1659051d58842d | Pending        | t2.micro      | -            | No alarms    | + us-east-2c      | -               |
| web       | i-01cb86bcf1fa144   | Running        | t2.micro      | -            | No alarms    | + us-east-2c      | -               |
| app2      | i-0b1f0945a7af737b5 | Stopped        | t2.micro      | -            | No alarms    | + us-east-2c      | -               |

The screenshot shows the AWS Management Console with the EC2 service selected. The main page displays a summary of the instance configuration, including the name 'db-2', the software image (Amazon Linux 2023 AMI), and the virtual server type (t2.micro). A tooltip provides information about the free tier. At the bottom right, there are 'Launch instance' and 'Review commands' buttons.

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.2.2...  
Virtual server type (instance type): t2.micro

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

Launch instance

The screenshot shows the detailed view of the selected Amazon Linux 2023 AMI. It includes the AMI ID (ami-06d4b7182ac3480fa), architecture (64-bit (x86)), and AMI ID (ami-06d4b7182ac3480fa). A tooltip provides information about the free tier. At the bottom right, there are 'Launch instance' and 'Review commands' buttons.

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.2.2...  
Virtual server type (instance type): t2.micro

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

Launch instance

Description  
Amazon Linux 2023 AMI 2023.2.20231113.0 x86\_64 HVM kernel-6.1

Architecture  
64-bit (x86)    AMI ID  
ami-06d4b7182ac3480fa    Verified provider

Instance type  
t2.micro    Free tier eligible  
Family: t2 1 vCPU 1 GiB Memory Current generation: true  
On-Demand Linux base pricing: 0.0116 USD per Hour  
On-Demand SUSE base pricing: 0.0116 USD per Hour  
On-Demand Windows base pricing: 0.0162 USD per Hour  
On-Demand RHEL base pricing: 0.0116 USD per Hour

Additional costs apply for AMIs with pre-installed software

Key pair (login)  
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  
newkey    Create new key pair

Summary  
Number of instances: 1

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.2.2...read more  
ami-06d4b7182ac3480fa

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 year.

Cancel    Launch instance    Review commands

Instance type  
t2.micro    Free tier eligible  
Family: t2 1 vCPU 1 GiB Memory Current generation: true  
On-Demand Linux base pricing: 0.0116 USD per Hour  
On-Demand SUSE base pricing: 0.0116 USD per Hour  
On-Demand Windows base pricing: 0.0162 USD per Hour  
On-Demand RHEL base pricing: 0.0116 USD per Hour

Additional costs apply for AMIs with pre-installed software

Key pair (login)  
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  
newkey    Create new key pair

Network settings  
Edit  
Network Info  
vpc-0f4523b2ab35df0f8  
Subnet Info  
No preference (Default subnet in any availability zone)  
Auto-assign public IP Info  
Enabled

Summary  
Number of instances: 1

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.2.2...read more  
ami-06d4b7182ac3480fa

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 year.

Cancel    Launch instance    Review commands



The screenshot shows the AWS EC2 Instances launch success page. A green success banner at the top states: "Success Successfully initiated launch of instance (i-049a565ac93c80628)". Below the banner, there's a "Launch log" button. Under "Next Steps", there are four cards: "Create billing and free tier usage alerts", "Connect to your instance", "Connect an RDS database", and "Create EBS snapshot policy". Each card has a descriptive text, a call-to-action button, and a "Learn more" link.

The screenshot shows the AWS EC2 Instances dashboard. On the left, a sidebar navigation includes "EC2 Dashboard", "EC2 Global View", "Events", "Instances" (selected), "Instance Types", "Launch Templates", "Spot Requests", "Savings Plans", "Reserved Instances", "Dedicated Hosts", "Capacity Reservations" (New), "Images" (AMIs, AMI Catalog), "Elastic Block Store" (Volumes, Snapshots, Lifecycle Manager), and "Network & Security". The main area displays a table titled "Instances (1/7) Info" with the following data:

| Name         | Instance ID         | Instance state | Instance type | Status check      | Alarm status | Availability Zone | Public IPv4 DNS |
|--------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|-----------------|
| db-2         | i-049a565ac93c80628 | Running        | t2.micro      | Initializing      | No alarms    | us-east-2c        | -               |
| db           | i-0cff86cbc1f1fa144 | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -               |
| <b>web-2</b> | i-08a1659031d58842d | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -               |
| web          | i-017cb989e00cf53f4 | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -               |
| app2         | i-0bf10945a7af737b5 | Stopped        | t2.micro      | -                 | No alarms    | us-east-2c        | -               |

Below the table, a detailed view for instance "i-08a1659031d58842d (web-2)" is shown with tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. The "Details" tab displays the instance summary, including Instance ID, Public IPv4 address (18.222.0.156), Private IPv4 address (10.1.3.86), Instance state (Running), and IP name (ip-10-1-3-86.us-east-2.compute.internal).

The screenshot shows the AWS EC2 Security Groups page for the 'sg-05d1781b75052f6df - default' group. The left sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, and Network & Security.

**Details**

|                     |                      |                            |                       |
|---------------------|----------------------|----------------------------|-----------------------|
| Security group name | Security group ID    | Description                | VPC ID                |
| default             | sg-05d1781b75052f6df | default VPC security group | vpc-0a6057a365f2533c8 |
| Owner               | Inbound rules count  | Outbound rules count       |                       |
| 626130759947        | 1 Permission entry   | 1 Permission entry         |                       |

**Inbound rules (1)**

| Name | Security group rule... | IP version | Type        | Protocol | Port range |
|------|------------------------|------------|-------------|----------|------------|
| -    | sgr-0d5252c17c8d82d... | -          | All traffic | All      | All        |

**Edit inbound rules**

Inbound rules control the incoming traffic that's allowed to reach the instance.

**Inbound rules**

| Security group rule ID | Type        | Protocol | Port range | Source  | Description |
|------------------------|-------------|----------|------------|---------|-------------|
| -                      | All traffic | All      | All        | Anyw... | 0.0.0.0/0   |

**Warning:** Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Buttons: Cancel, Preview changes, Save rules.

Screenshot of the AWS Cloud Console showing the EC2 Instances page. The browser tab is "Instances | EC2" and the URL is "us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#InstancesInstanceState=running".

The left sidebar shows navigation links for EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security.

The main content area displays a table titled "Instances (1/4) Info" with the following columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. There are four instances listed:

| Name         | Instance ID                | Instance state | Instance type | Status check      | Alarm status | Availability Zone | Public IPv4 DNS |
|--------------|----------------------------|----------------|---------------|-------------------|--------------|-------------------|-----------------|
| db-2         | i-049a565ac93c80628        | Running        | t2.micro      | Initializing      | No alarms    | us-east-2c        | -               |
| db           | i-0ccfb86cbc1fa144         | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -               |
| <b>web-2</b> | <b>i-08a1659031d58842d</b> | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -               |
| web          | i-017cb989e00cf53f4        | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -               |

Below the table, a modal window is open for the instance "web-2" (i-08a1659031d58842d). The "Details" tab is selected, showing the following details:

- Instance summary**: Instance ID: i-08a1659031d58842d (web-2), Public IPv4 address: 18.222.0.156, Instance state: Running, Private IP DNS name (IPv4 only): ip-10-1-3-86.us-east-2.compute.internal.
- Networking**: Subnet: subnet-00000000, VPC: vpc-00000000, Security group: sg-00000000.
- Storage**: Volume: /dev/sda1, Size: 20 GiB, Type: Standard SSD.
- Status checks**: 2/2 checks passed.
- Monitoring**: CloudWatch Metrics: Enabled, CloudWatch Logs: Enabled.
- Tags**: None.

At the bottom right of the modal, there are links for "Edit instance details", "View logs", and "View metrics".

Screenshot of the AWS Cloud Console showing the EC2 Instances page. The browser tab is "Instances | EC2" and the URL is "us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#InstancesInstanceState=running".

The left sidebar shows navigation links for EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security.

The main content area displays a table titled "Instances (1/4) Info" with the following columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Actions, and Launch instances. There are four instances listed:

| Name         | Instance ID                | Instance state | Instance type | Status check      | Alarm status | Actions                      | Launch instances                         |
|--------------|----------------------------|----------------|---------------|-------------------|--------------|------------------------------|------------------------------------------|
| db-2         | i-049a565ac93c80628        | Running        | t2.micro      | Initializing      | No alarms    | <a href="#">Connect</a>      | <a href="#">View details</a>             |
| db           | i-0ccfb86cbc1fa144         | Running        | t2.micro      | 2/2 checks passed | No alarms    | <a href="#">View details</a> | <a href="#">Security</a>                 |
| <b>web-2</b> | <b>i-08a1659031d58842d</b> | Running        | t2.micro      | 2/2 checks passed | No alarms    | <a href="#">View details</a> | <a href="#">Image and templates</a>      |
| web          | i-017cb989e00cf53f4        | Running        | t2.micro      | 2/2 checks passed | No alarms    | <a href="#">View details</a> | <a href="#">Monitor and troubleshoot</a> |

Below the table, a modal window is open for the instance "web-2" (i-08a1659031d58842d). The "Details" tab is selected, showing the following details:

- Instance summary**: Instance ID: i-08a1659031d58842d (web-2), Public IPv4 address: 18.222.0.156, Instance state: Running, Private IP DNS name (IPv4 only): ip-10-1-3-86.us-east-2.compute.internal.
- Networking**: Subnet: subnet-00000000, VPC: vpc-00000000, Security group: sg-00000000.
- Storage**: Volume: /dev/sda1, Size: 20 GiB, Type: Standard SSD.
- Status checks**: 2/2 checks passed.
- Monitoring**: CloudWatch Metrics: Enabled, CloudWatch Logs: Enabled.
- Tags**: None.

At the bottom right of the modal, there are links for "Edit instance details", "View logs", and "View metrics".

The screenshot shows the AWS EC2 Instances page with the instance `i-08a1659031d58842d` selected. The `Connect to instance` section is open, displaying two main connection methods:

- EC2 Instance Connect**: Selected.
- Connect using EC2 Instance Connect Endpoint**

Below these options, the public IP address is listed as `18.222.0.156`. The user name is set to `ec2-user`. A note at the bottom states: "Note: In most cases, the default user name, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name."

The screenshot shows an EC2 serial console session for the instance `i-08a1659031d58842d`. The terminal window displays the following output:

```

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-1-3-86 ~]$ ping google.com
PING google.com (172.217.1.110) 56(84) bytes of data.
64 bytes from yzz08s09-in-f110.1e100.net (172.217.1.110): icmp_seq=1 ttl=55 time=16.5 ms
64 bytes from yzz08s09-in-f14.1e100.net (172.217.1.110): icmp_seq=2 ttl=55 time=16.6 ms
64 bytes from ord37s51-in-f14.1e100.net (172.217.1.110): icmp_seq=3 ttl=55 time=16.6 ms
64 bytes from mia09s17-in-f14.1e100.net (172.217.1.110): icmp_seq=4 ttl=55 time=16.6 ms
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 16.490/16.585/16.628/0.055 ms
[ec2-user@ip-10-1-3-86 ~]$

```

At the bottom of the terminal window, it says `i-08a1659031d58842d (web-2)`, `PublicIPs: 18.222.0.156`, and `PrivateIPs: 10.1.3.86`.

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like EC2 Dashboard, Services, and Instances. The Instances section is expanded, showing sub-options like Instances, Instance Types, Launch Templates, etc. The main content area displays a table of instances:

| Name         | Instance ID                | Instance state | Instance type   | Status check             | Alarm status     | Availability Zone | Public IPv4 |
|--------------|----------------------------|----------------|-----------------|--------------------------|------------------|-------------------|-------------|
| db-2         | i-049a565ac95c80628        | Running        | t2.micro        | 2/2 checks passed        | No alarms        | us-east-2c        | -           |
| db           | i-0ccfb86cbc1fa144         | Running        | t2.micro        | 2/2 checks passed        | No alarms        | us-east-2c        | -           |
| <b>web-2</b> | <b>i-08a1659031d58842d</b> | <b>Running</b> | <b>t2.micro</b> | <b>2/2 checks passed</b> | <b>No alarms</b> | <b>us-east-2c</b> | <b>-</b>    |
| web          | i-017cb989e00cf53f4        | Running        | t2.micro        | 2/2 checks passed        | No alarms        | us-east-2c        | -           |

Below the table, a modal window is open for the selected instance 'web-2'. It shows details like Public IPv4 address (18.222.0.156), Instance state (Running), and Private IP DNS name (ip-10-1-3-86.us-east-2.compute.internal).

The screenshot shows the 'Connect to instance' dialog. At the top, it says 'Connect to your instance i-08a1659031d58842d (web-2) using any of these options'. Below this, there are four tabs: EC2 Instance Connect, Session Manager, SSH client, and EC2 serial console. The 'EC2 Instance Connect' tab is selected.

The 'Connection Type' section has two options: 'Connect using EC2 Instance Connect' (selected) and 'Connect using EC2 Instance Connect Endpoint'. Both options have sub-instructions. The 'Public IP address' field contains '18.222.0.156'. The 'User name' field contains 'ec2-user'. A note at the bottom of this section says: 'Note: In most cases, the default user name, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.'

At the bottom right of the dialog are 'Cancel' and 'Connect' buttons.

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec 3 11:10:46 2023 from 3.16.146.5
[ec2-user@ip-10-1-3-86 ~]$ ping 10.1.18.132
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.
64 bytes from 10.1.18.132: icmp_seq=1 ttl=127 time=0.806 ms
64 bytes from 10.1.18.132: icmp_seq=2 ttl=127 time=0.498 ms
64 bytes from 10.1.18.132: icmp_seq=3 ttl=127 time=0.449 ms
64 bytes from 10.1.18.132: icmp_seq=4 ttl=127 time=0.566 ms
64 bytes from 10.1.18.132: icmp_seq=5 ttl=127 time=0.465 ms

i-08a1659031d58842d (web-2)
PublicIPs: 18.222.0.156 PrivateIPs: 10.1.3.86
```

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec 3 11:10:46 2023 from 3.16.146.5
[ec2-user@ip-10-1-3-86 ~]$ ping 10.1.18.132
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.
64 bytes from 10.1.18.132: icmp_seq=1 ttl=127 time=0.806 ms
64 bytes from 10.1.18.132: icmp_seq=2 ttl=127 time=0.498 ms
64 bytes from 10.1.18.132: icmp_seq=3 ttl=127 time=0.449 ms
64 bytes from 10.1.18.132: icmp_seq=4 ttl=127 time=0.566 ms
64 bytes from 10.1.18.132: icmp_seq=5 ttl=127 time=0.465 ms
64 bytes from 10.1.18.132: icmp_seq=6 ttl=127 time=0.494 ms
64 bytes from 10.1.18.132: icmp_seq=7 ttl=127 time=0.533 ms
^C
--- 10.1.18.132 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6270ms
rtt min/avg/max/mdev = 0.449/0.544/0.806/0.112 ms
[ec2-user@ip-10-1-3-86 ~]$ ping 10.0.63.146
PING 10.0.63.146 (10.0.63.146) 56(84) bytes of data.
```

```

newkey
File Edit View
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQeAk0v9A01waMcVqB5MgUNlRM/bf7x12pnVozQy0HPXisBkwj
4BgbV4kqSre7ez/SIV2vza6zGACNT-Mxb0tBTRzV2I63/7DGbnbgL5qREAbqyKz
S7tr1giuso8nmWIM1RUUIxie30k/TplxFbcIC1y0p2rPTYD07FxCy9xpkB1j
je9jjayw11ss3mQ5rm8nJ9qJDFNKU69XQ0QiyAumqv+Q6FBgZg79/m166d24+zT
E580QUxhPHq7u05FKNR8Nugnzs1jM6g803h1wfyDyGhbdRMw/mm/1quz/JyW5
yrADABAYoWejba0a7LPNa102TwLu0xygHDx8hW1DAQABAOIBAC2FwCc6ekv4k7z
TadWbSKFvA28LH1+zhkfFq7+AlEiWkyoS1s929+r9kVNSaV7zjtv10bo
f1im0u7vONwagjnNmVyaFCAggB1B8uggp0lQHcWHAjZUpLj0uNnluHeOMRwf
H4sqIpohgK3Cg19ZDwTdwR7XGmdwALMKObw9g/ga13NaW8+06fxCx7E4jFBv
Qbjjashofx0j0e8Shn9a5nAB1MhdBhfKA/Ua9adb19fnSSyvg9Kpd1sk6cjt7A
A3ly/y7Kgsx1AwvfINTIZa1hnd49fnc2T1991s+GHln6x3b+Hk96u52Bsg1txDk
Gv1wzECgYEAgv8moe+6H8KLpZ8Xdon5e4vXDE0kV8uH1Z1LEnvV1yx0kFm04Mz
ULN7P070j+itb19mz2ZgbTQ7jhvn3w0tDw0Bamxju0qKnGhTxseR7MyioQ
JSYNBVF15MLMc/WCEG0mjB8u9Op0p5PvcAtJnp+10gVnQxy+p-/uLMcgYEAoHC/
PdW71wNwHs9yWEY2/2d0Y-FDokzf9aeoES/ihyLcceB2pMq2NLc5LtDRx1av
g56kb71jk1KdfEfwsjF4KEBT0Ty126tandjCambTqk10hCNqr/RnsCduEch
qHkm+v1XlF4uHEQ0Rt/77Q14Lfm1ehoGzrYUcgeYEAS8r1k+ZvFFvFzLxeBR3a
wv537RLEbdlbxqOKw+98Xu+exdd/knglmhkZG/TS1Fnjd2vFrVxIn5B8mqgrGtpe
ORJAQf3B7B8+n/sL00B+8e10HKXPwCe2Aal8ym51jpkkaQ0uyLuK0Uz5d+80
T4MvhkzJ8160PQ05LrPj0u0gYBuzE4V2uOYsPcsuA0TLY1aP0zmsPsvjCqjn0T
P22pZ08tucJfxM7e8QoSxcdNaKt2VTV/GoF/eB04+6jkdj7y2nCLGA3tD12
RjTR5qX0pxcmju08D7ghDp2j31012t3DXkksj0MKm+1f2w+3t/1704BgZRMF82y1
F3pxQKgB40Exu0209pramG1ivx3jHyDefgfw/640LQLbxVytKu90+pangOXI4cdOfVfiw7LfyDw0+Hu0wl871LBTYGw4Rmb3FztBnJppzovfBuXW3
DARuhufewepcb3Naqj/buuy1uhxs7zyNz3HxLd06EA4Lf0/op
-----END RSA PRIVATE KEY-----

```

i-08a1659031d58842d (web-2)  
Public IPs: 18.222.0.156 Private IPs: 10.1.3.86

| Name      | Instance ID                | Instance state | Instance type   | Status check             | Alarm status     | Availability Zone | Public IP |
|-----------|----------------------------|----------------|-----------------|--------------------------|------------------|-------------------|-----------|
| web-2     | i-08a1659031d58842d        | Running        | t2.micro        | 2/2 checks passed        | No alarms        | us-east-2c        | -         |
| db-2      | i-049a565ac93c80628        | Running        | t2.micro        | 2/2 checks passed        | No alarms        | us-east-2c        | -         |
| <b>db</b> | <b>i-0ccfb86cbcf1fa144</b> | <b>Running</b> | <b>t2.micro</b> | <b>2/2 checks passed</b> | <b>No alarms</b> | <b>us-east-2c</b> | <b>-</b>  |
| web       | i-017cb989e00cf53f4        | Running        | t2.micro        | 2/2 checks passed        | No alarms        | us-east-2c        | -         |

**Instance: i-0ccfb86cbcf1fa144 (db)**

| Details                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Security                                                                     | Networking                            | Storage | Status checks | Monitoring | Tags |                                         |                          |                                       |                   |                           |                      |                                                                     |                                                                              |  |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|---------------------------------------|---------|---------------|------------|------|-----------------------------------------|--------------------------|---------------------------------------|-------------------|---------------------------|----------------------|---------------------------------------------------------------------|------------------------------------------------------------------------------|--|
| <b>Instance summary</b> <a href="#">Info</a> <table border="1"> <tr> <td>Instance ID<br/>i-0ccfb86cbcf1fa144 (db)</td> <td>Public IPv4 address<br/>-</td> <td>Private IPv4 addresses<br/>10.0.63.146</td> </tr> <tr> <td>IPv6 address<br/>-</td> <td>Instance state<br/>Running</td> <td>Public IPv4 DNS<br/>-</td> </tr> <tr> <td>Hostname type<br/>IP name: ip-10-0-63-146.us-east-2.compute.internal</td> <td colspan="2">Private IP DNS name (IPv4 only)<br/>ip-10-0-63-146.us-east-2.compute.internal</td> </tr> </table> |                                                                              |                                       |         |               |            |      | Instance ID<br>i-0ccfb86cbcf1fa144 (db) | Public IPv4 address<br>- | Private IPv4 addresses<br>10.0.63.146 | IPv6 address<br>- | Instance state<br>Running | Public IPv4 DNS<br>- | Hostname type<br>IP name: ip-10-0-63-146.us-east-2.compute.internal | Private IP DNS name (IPv4 only)<br>ip-10-0-63-146.us-east-2.compute.internal |  |
| Instance ID<br>i-0ccfb86cbcf1fa144 (db)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Public IPv4 address<br>-                                                     | Private IPv4 addresses<br>10.0.63.146 |         |               |            |      |                                         |                          |                                       |                   |                           |                      |                                                                     |                                                                              |  |
| IPv6 address<br>-                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Instance state<br>Running                                                    | Public IPv4 DNS<br>-                  |         |               |            |      |                                         |                          |                                       |                   |                           |                      |                                                                     |                                                                              |  |
| Hostname type<br>IP name: ip-10-0-63-146.us-east-2.compute.internal                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Private IP DNS name (IPv4 only)<br>ip-10-0-63-146.us-east-2.compute.internal |                                       |         |               |            |      |                                         |                          |                                       |                   |                           |                      |                                                                     |                                                                              |  |

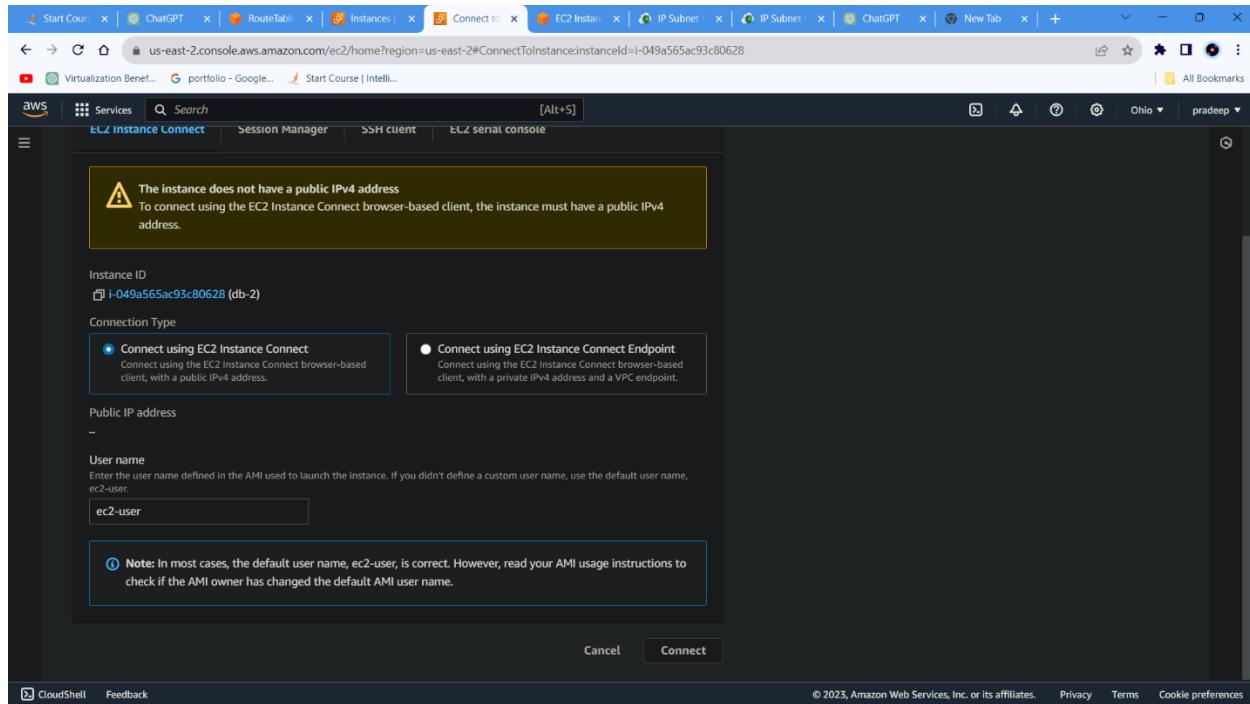
The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 Dashboard, EC2 Global View, Events, Instances (with sub-options for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security. The main content area displays a table of instances:

| Name  | Instance ID          | Instance state | Instance type | Status check      | Alarm status | Availability Zone | Public IP |
|-------|----------------------|----------------|---------------|-------------------|--------------|-------------------|-----------|
| web-2 | i-08a1659031d58842d  | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -         |
| db-2  | i-049a565ac95c80628  | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -         |
| db    | i-0ccfb86cbcffffa144 | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -         |
| web   | i-017cb989e00cf53f4  | Running        | t2.micro      | 2/2 checks passed | No alarms    | us-east-2c        | -         |

Below the table, a modal window is open for the instance **i-049a565ac95c80628 (db-2)**. The modal has tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. The Details tab is selected, showing the following information:

| Instance ID                                        | Public IPv4 address                       | Private IPv4 addresses |
|----------------------------------------------------|-------------------------------------------|------------------------|
| i-049a565ac95c80628 (db-2)                         | -                                         | 10.1.18.132            |
| IPv6 address                                       | Instance state                            | Public IPv4 DNS        |
| -                                                  | Running                                   | -                      |
| Hostname type                                      | Private IP DNS name (IPv4 only)           | Instance type          |
| IP name: ip-10-1-18-132.us-east-2.compute.internal | ip-10-1-18-132.us-east-2.compute.internal | t2.micro               |

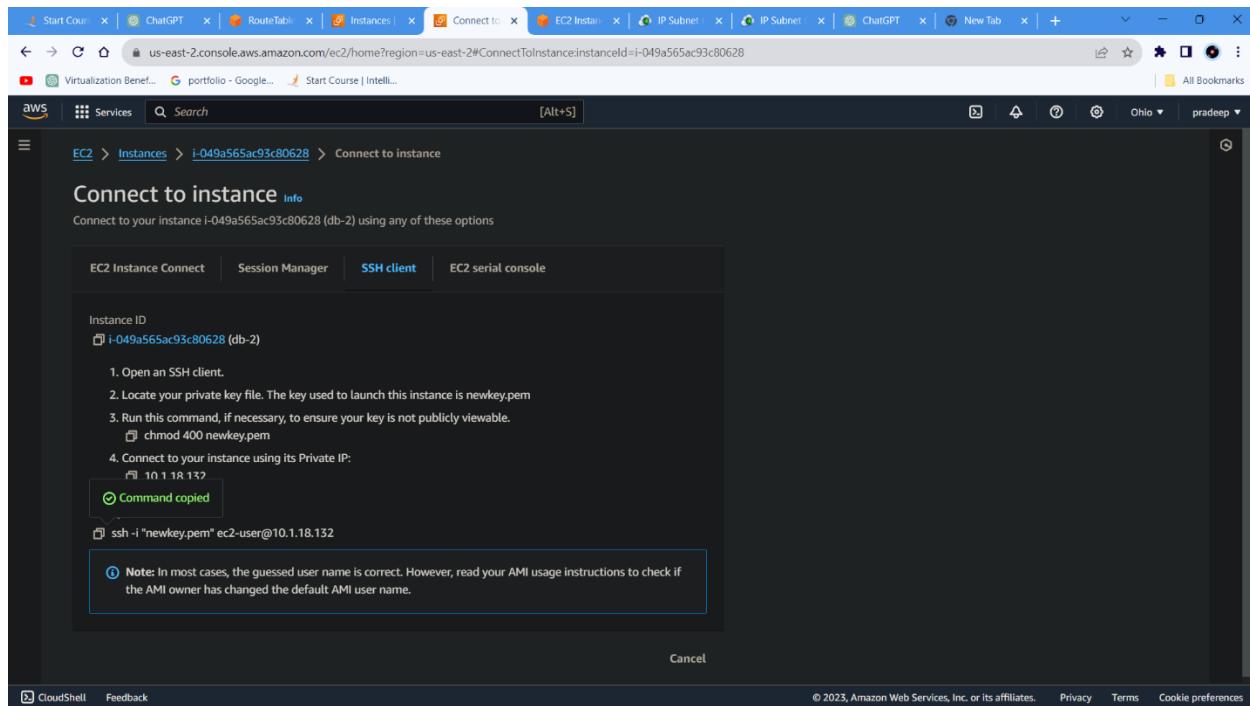
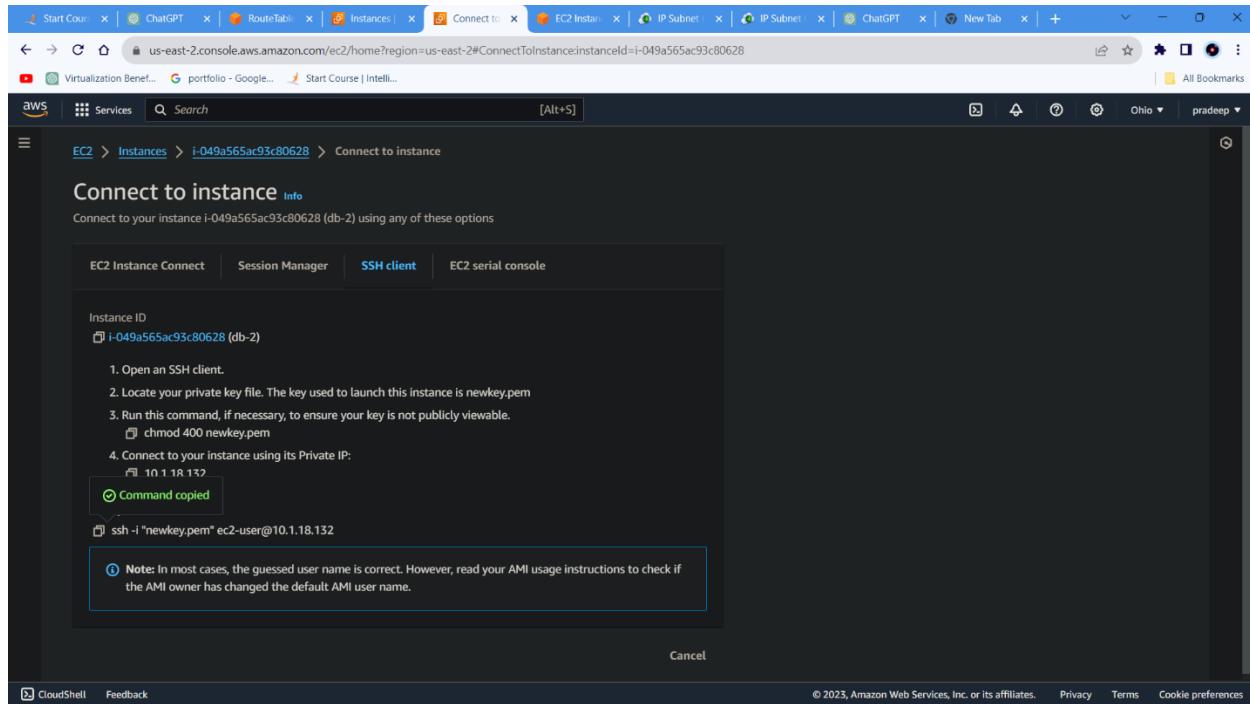
This screenshot is identical to the one above, but the Actions dropdown menu is open over the row for the **db-2** instance. The dropdown menu includes options: Connect, View details, Manage instance state, Instance settings, Networking, Security, Image and templates, and Monitor and troubleshoot.



```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec 3 11:17:48 2023 from 3.16.146.3
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
is and I
[ec2-user@ip-10-1-3-86 ~]$ sudo chmod 400 key.pem
```

i-08a1659031d58842d (web-2)  
PublicIPs: 18.222.0.156 PrivateIPs: 10.1.3.86



The screenshot shows a browser window with multiple tabs open. The active tab is titled "EC2 Instance Connect" and displays a terminal session on an Amazon Linux 2023 instance. The terminal output shows the user attempting to connect to another host via SSH, but the connection fails due to a known host fingerprint mismatch.

```
Last login: Sun Dec 3 11:17:48 2023 from 3.16.146.3
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo ssh -i "key.pem" ec2-user@10.1.18.132
The authenticity of host '10.1.18.132 (10.1.18.132)' can't be established.
ED25519 key fingerprint is SHA256:v44yMCXgWP5TYO1l0tW36EyaRCNoZZj2StAznWahs6M.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/(fingerprint))? yes
```

i-08a1659031d58842d (web-2)  
PublicIPs: 18.222.0.156 PrivateIPs: 10.1.3.86

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This screenshot shows the same AWS CloudShell session after the user has added the host's fingerprint to their known hosts. The terminal now successfully connects to the remote host without prompting for confirmation.

```
Last login: Sun Dec 3 11:17:48 2023 from 3.16.146.3
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo ssh -i "key.pem" ec2-user@10.1.18.132
The authenticity of host '10.1.18.132 (10.1.18.132)' can't be established.
ED25519 key fingerprint is SHA256:v44yMCXgWP5TYO1l0tW36EyaRCNoZZj2StAznWahs6M.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/(fingerprint))? yes
Warning: Permanently added '10.1.18.132' (ED25519) to the list of known hosts.
[ec2-user@ip-10-1-18-132 ~]$
```

i-08a1659031d58842d (web-2)  
PublicIPs: 18.222.0.156 PrivateIPs: 10.1.3.86

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the 'Connect to instance' dialog box for an EC2 instance with ID i-0ccfb86cbcf1fa144. The dialog includes tabs for 'EC2 Instance Connect', 'Session Manager', 'SSH client' (which is selected), and 'EC2 serial console'. It provides instructions for connecting via SSH, including steps to open an SSH client, locate a private key file named newkey.pem, run chmod 400 on it, and connect to the instance's private IP 10.0.63.146. An example command is shown: ssh -i "newkey.pem" ec2-user@10.0.63.146. A note cautions users to verify the AMI owner has not changed the default AMI user name.

The screenshot shows the EC2 Instances page with 7 instances listed. The instance i-0ccfb86cbcf1fa144 (db) is selected. The details pane for this instance shows the following information:

| Attribute                       | Value                                     |
|---------------------------------|-------------------------------------------|
| Instance ID                     | i-0ccfb86cbcf1fa144 (db)                  |
| Public IPv4 address             | -                                         |
| Private IPv4 address            | 10.0.63.146                               |
| Instance state                  | Running                                   |
| Private IP DNS name (IPv4 only) | ip-10-0-63-146.us-east-2.compute.internal |

**EC2 > Instances > i-0ccfb86cbcf1fa144 > Connect to instance**

**Connect to instance** Info

Connect to your instance i-0ccfb86cbcf1fa144 (db) using any of these options

EC2 Instance Connect Session Manager **SSH client** EC2 serial console

Instance ID  
i-0ccfb86cbcf1fa144 (db)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is newkey.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.  
 chmod 400 newkey.pem
4. Connect to your instance using its Private IP:  
 10.0.63.146

Example:  
 ssh -i "newkey.pem" ec2-user@10.0.63.146

**Note:** In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

**Cancel**

```

>Last login: Sun Dec 3 11:17:48 2023 from 3.16.146.3
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo ssh -i "key.pem" ec2-user@10.1.18.132
The authenticity of host '10.1.18.132 (10.1.18.132)' can't be established.
ED25519 key fingerprint is SHA256:v4YMCXgWP5tYOIIot3W6EyaRCN0zzj2StAznWahs6M.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.1.18.132' (ED25519) to the list of known hosts.
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-10-1-18-132 ~]$ ping 10.0.63.146

```

i-08a1659031d58842d (web-2)  
PublicIPs: 18.222.0.156 PrivateIPs: 10.1.3.86

```

Last login: Sun Dec 3 11:17:48 2023 from 3.16.146.3
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ sudo nano key.pem
[ec2-user@ip-10-1-3-86 ~]$ ls
[ec2-user@ip-10-1-3-86 ~]$ key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-1-3-86 ~]$ sudo ssh -i "key.pem" ec2-user@10.1.18.132
The authenticity of host '10.1.18.132 (10.1.18.132)' can't be established.
ED25519 key fingerprint is SHA256:v44yMCXgWP5TYO1IoT3W6EyaRCN0zzj2StAznWahs6M.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/(fingerprint))? yes
Warning: Permanently added '10.1.18.132' (ED25519) to the list of known hosts.

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-1-18-132 ~]$ ping 10.0.63.146
PING 10.0.63.146 (10.0.63.146) 56(84) bytes of data.

```

i-08a1659031d58842d (web-2)  
PublicIPs: 18.222.0.156 PrivateIPs: 10.1.3.86

Route table ID  
rtb-0a42d02d945304778

VPC  
vpc-0a6057a365f2533c8 | development-network-vpc

| Routes                             | Subnet associations | Edge |
|------------------------------------|---------------------|------|
| <b>Routes (2)</b>                  |                     |      |
| <input type="text"/> Filter routes |                     |      |

- gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- Endpoints
- Endpoint services
- NAT gateways
- Peering connections

▼ Security

The screenshot shows the AWS VPC Peering Connections page. The left sidebar is expanded, showing sections like Virtual private cloud, Security, and VPC Firewall. Under Peering connections, it lists 'my-peering'. The main content area is titled 'Peering connections' and shows a table with one row: 'No peering connection found'. A message below the table says 'Select a peering connection above'.

The screenshot shows the 'Create peering connection' wizard. Step 1: Peering connection settings. It asks for a name (optional) and selects 'my-peering'. It then asks to 'Select a local VPC to peer with' and shows a dropdown with 'VPC ID (Requester)' set to 'vpc-0a6057a365f2533c8 (development-network-vpc)'. Below this, it lists 'VPC CIDRs for vpc-0a6057a365f2533c8 (development-network-vpc)' with a table showing one entry: 'CIDR: 10.1.0.0/16, Status: Associated, Status reason: -'. The next section, 'Select another VPC to peer with', includes fields for Account ('My account' is selected) and Region ('This Region (us-east-2)' is selected).

Screenshot of the AWS VPC Peering Connection creation interface.

**Requester VPC:**

- VPC ID (Requester): vpc-0a6057a365f2533c8 (development-network-vpc)
- VPC CIDRs for vpc-0a6057a365f2533c8 (development-network-vpc):
 

| CIDR        | Status     | Status reason |
|-------------|------------|---------------|
| 10.1.0.0/16 | Associated | -             |

**Accepter VPC:**

- VPC ID (Acceptor): vpc-046c5e0929b626cc4 (Network-production-vpc)
- VPC CIDRs for vpc-046c5e0929b626cc4 (Network-production-vpc):
 

| CIDR        | Status     | Status reason |
|-------------|------------|---------------|
| 10.0.0.0/16 | Associated | -             |

**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.

CloudShell   Feedback   © 2023, Amazon Web Services, Inc. or its affiliates.   Privacy   Terms   Cookie preferences

Screenshot of the AWS VPC Peering Connection details page.

**Peering Connection Details:**

A VPC peering connection ppx-0b86eda95b59fc25c / my-peering has been requested.

**Actions:**

Pending acceptance

You can accept or reject this peering connection request using the 'Actions' menu. You have until Sunday, December 10, 2023 at 17:06:56 GMT+5:30 to accept or reject the request, otherwise it expires.

**Details:**

|                    |                                                |                   |                                                 |
|--------------------|------------------------------------------------|-------------------|-------------------------------------------------|
| Requester owner ID | 626130759947                                   | Acceptor owner ID | 626130759947                                    |
| Peer connection ID | ppx-0b86eda95b59fc25c                          | Requester VPC     | vpc-0a6057a365f2533c8 / development-network-vpc |
| Status             | Pending Acceptance by 626130759947             | Requester CIDRs   | 10.1.0.0/16                                     |
| Expiration time    | Sunday, December 10, 2023 at 17:06:56 GMT+5:30 | Requester Region  | Ohio (us-east-2)                                |
|                    |                                                | Acceptor VPC      | vpc-046c5e0929b626cc4 / Network-production-vpc  |
|                    |                                                | Acceptor CIDRs    | -                                               |
|                    |                                                | Acceptor Region   | Ohio (us-east-2)                                |

DNS   Route tables   Tags   Edit DNS settings

CloudShell   Feedback   © 2023, Amazon Web Services, Inc. or its affiliates.   Privacy   Terms   Cookie preferences

The screenshot shows the AWS VPC Peering Connections page. On the left, a navigation sidebar lists various VPC-related services. The main content area displays a table titled "Peering connections (1)". The table has columns for Name, Peering connection ID, Status, Requester VPC, and Acceptor VPC. A single row is shown for "my-peering" with the ID "pcx-0b86eda95b59fc25c". The status is "Pending acceptance". The Requester VPC is "vpc-0a6057a365f2533c8 / dev...". The Acceptor VPC is "vpc-046c5e0929b626cc4 / Net...". The "Actions" dropdown menu at the top right of the table includes options like "Create peering connection", "View details", "Accept request" (which is highlighted), "Reject request", "Edit DNS settings", "Manage tags", and "Delete peering connection".

This screenshot shows the detailed view for the "my-peering" peering connection. At the top, it says "Peering connections (1/1) Info". Below that is a table with the same columns as the previous screenshot. The "Actions" dropdown menu is open, with "Accept request" highlighted. A modal window titled "Pending acceptance" is displayed, stating: "You can accept or reject this peering connection request using the 'Actions' menu. You have until Sunday, December 10, 2023 at 17:06:56 GMT+5:30 to accept or reject the request, otherwise it expires." At the bottom of the page, there are tabs for "Details", "DNS", "Route tables", and "Tags". The "Details" tab is selected. It shows the Requester owner ID as "626130750047" and the Acceptor owner ID as "626130750047". The VPC Peering connection ARN is also listed.

The screenshot shows the AWS VPC Route Tables page. On the left, there's a navigation sidebar with sections like VPC dashboard, EC2 Global View, Filter by VPC (with a dropdown for 'Select a VPC'), Virtual private cloud (Your VPCs, Subnets, Route tables), Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections. Below that is a Security section for Network ACLs and Security groups. At the bottom of the sidebar are CloudShell and Feedback links. The main content area has a search bar and a table titled 'Route tables (6)'. The table columns are Name, Route table ID, Explicit subnet associations, Edge associations, Main, and VPC. The rows list various route tables with their corresponding IDs and associations. A 'Create route table' button is at the top right of the table area.

The screenshot shows the 'Create route table' wizard. The title is 'Create route table' with an 'Info' link. Below it is a brief description: 'A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.' The first step is 'Route table settings'. It has a 'Name - optional' field containing 'my-route-table-01', a 'VPC' dropdown labeled 'Select a VPC', and a 'Tags' section with a note about tags being optional labels for resources. There's also an 'Add new tag' button and a note that 50 more tags can be added. At the bottom are 'Cancel' and 'Create route table' buttons.

The screenshot shows the AWS VPC 'Create route table' page. In the 'Route table settings' section, a tag named 'peering-rt1' is added under 'Name'. The 'VPC' dropdown is set to 'vpc-046c5e0929b626cc4 (Network-production-vpc)'. Under 'Tags', a single tag 'Name: peering-rt1' is listed. At the bottom right, the 'Create route table' button is highlighted.

The screenshot shows the AWS VPC 'Edit routes' page for route table 'rtb-082267f759c154654'. It lists two routes: one for '10.0.0.0/16' targeting 'local' (status: Active, propagated: No), and another for '10.1.16.0/20' targeting a 'Peering Connection' (status: -, propagated: No). A new route entry 'Add route' is visible at the bottom left. At the bottom right, the 'Save changes' button is highlighted.

Screenshot of the AWS VPC Route Tables page.

**Route tables (1/7) Info**

| Name                  | Route table ID               | Explicit subnet associations | Edge associations |
|-----------------------|------------------------------|------------------------------|-------------------|
| rtb-0a44f83d9f1b19a2d | -                            | -                            | -                 |
| Nat-RT                | rtb-0f8d16e7043da9fc         | 2 subnets                    | -                 |
| -                     | rtb-09d3f55c23e268461        | -                            | -                 |
| network-production-RT | rtb-046ac920c29cd79ee        | subnet_04e89e4ee3018b...     | -                 |
| -                     | rtb-072954dc95ffa65cc        | -                            | -                 |
| development -RT       | rtb-0a42d02d945304778        | subnet_0a77f803c38578...     | -                 |
| <b>peering-rt1</b>    | <b>rtb-082267f759c154654</b> | -                            | -                 |

**rtb-082267f759c154654 / peering-rt1**

**Details**

|                                            |                                                |                                   |                        |
|--------------------------------------------|------------------------------------------------|-----------------------------------|------------------------|
| Route table ID<br>rtb-082267f759c154654    | Main<br><input checked="" type="checkbox"/> No | Explicit subnet associations<br>- | Edge associations<br>- |
| VPC<br>vpc-046c5e0929b626cc4   Network-... | Owner ID<br>626130759947                       |                                   |                        |

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS VPC Edit Route Table Subnet Associations page.

**Edit subnet associations**

Change which subnets are associated with this route table.

**Available subnets (1/5)**

| Name     | Subnet ID                | IPv4 CIDR    | IPv6 CIDR | Route table ID                        |
|----------|--------------------------|--------------|-----------|---------------------------------------|
| db       | subnet-01814985b7e28d2cc | 10.0.48.0/20 | -         | Main (rtb-072954dc95ffa65cc)          |
| db-cache | subnet-0a6d37403fd8d39e5 | 10.0.32.0/20 | -         | rtb-0f8d16e7043da9fc / Nat-RT         |
| app1     | subnet-0b13aa0d4eb8c74a9 | 10.0.0.0/20  | -         | rtb-0f8d16e7043da9fc / Nat-RT         |
| web      | subnet-04e89e4ee3018b76c | 10.0.64.0/20 | -         | rtb-046ac920c29cd79ee / network-pro.. |
| app2     | subnet-06f00917996347439 | 10.0.16.0/20 | -         | Main (rtb-072954dc95ffa65cc)          |

**Selected subnets**

- subnet-01814985b7e28d2cc / db

**Save associations**

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You have successfully updated subnet associations for rtb-082267f759c154654 / peering-rt1.

| Name                  | Route table ID        | Explicit subnet associations | Edge associations | Main | VPC                   |
|-----------------------|-----------------------|------------------------------|-------------------|------|-----------------------|
| rtb-082267f759c154654 | rtb-082267f759c154654 | subnet-01814985b7e28d...     | -                 | No   | vpc-046c5e0929b626cc4 |
| rtb-09d33f5523e268461 | rtb-09d33f5523e268461 | -                            | -                 | Yes  | vpc-0a6057a365f2533c8 |
| -                     | rtb-046ac920c29cd79ee | subnet-04e89e4ee3018b...     | -                 | No   | vpc-046c5e0929b626cc4 |
| network-production-RT | rtb-046ac920c29cd79ee | -                            | -                 | No   | vpc-046c5e0929b626cc4 |
| -                     | rtb-072954dc95ffa65cc | -                            | -                 | Yes  | vpc-046c5e0929b626cc4 |
| development -RT       | rtb-0a42d02d945304778 | subnet-0a77f803c39578...     | -                 | No   | vpc-0a6057a365f2533c8 |
| peering-rt1           | rtb-082267f759c154654 | subnet-01814985b7e28d...     | -                 | No   | vpc-046c5e0929b626cc4 |

**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="Name"/> | <input type="text" value="peering-rt-vpc2"/> |

**Add new tag**  
You can add 49 more tags.

**Create route table**

The screenshot shows the AWS VPC Route Table Details page. The URL is <https://us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#RouteTableDetails:RouteTableId=rtb-0d55613ea4275871a>. The page displays a success message: "Route table rtb-0d55613ea4275871a | peering-rt-vpc2 was created successfully." The left sidebar shows the VPC navigation path: VPC > Route tables > rtb-0d55613ea4275871a. The main content area shows the route table details, including its ID (rtb-0d55613ea4275871a), VPC (vpc-0a6057a365f253c8 | development-network-vpc), and a single route entry for destination 10.1.0.0/16 with target local and status Active.

The screenshot shows the AWS VPC Edit Routes page. The URL is <https://us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2>EditRoutes:RouteTableId=rtb-0d55613ea4275871a>. The page title is "Edit routes". It shows a table with columns: Destination, Target, Status, and Propagated. There are two rows: one for destination 10.1.0.0/16 with target local and status Active (No propagation), and another for destination 10.0.48.0/20 with target Peering Connection (pxc-0b86eda95b59fc25) and status - (No propagation). A "Remove" button is visible next to the second row. At the bottom, there are "Add route", "Cancel", "Preview", and "Save changes" buttons.

Updated routes for rtb-0d55613ea4275871a / peering-rt-vpc2 successfully

**rtb-0d55613ea4275871a / peering-rt-vpc2**

| Destination  | Target                | Status | Propagated |
|--------------|-----------------------|--------|------------|
| 10.0.48.0/20 | pcc-0b86eda95b59fc25c | Active | No         |
| 10.1.0.0/16  | local                 | Active | No         |

Route tables (1/8)

| Name                   | Route table ID               | Explicit subnet associations | Edge associations |
|------------------------|------------------------------|------------------------------|-------------------|
| development -RT        | rtb-0a42d02d945304778        | subnet-0a77f803c38578...     | -                 |
| <b>peering-rt-vpc2</b> | <b>rtb-0d55613ea4275871a</b> | -                            | -                 |
| Nat-RT                 | rtb-0f8d16e7043da9fc         | 2 subnets                    | -                 |
| peering-rt1            | rtb-082267f759c154654        | subnet-01814985b7e28d...     | -                 |
| -                      | rtb-09d33f5523e268461        | -                            | -                 |
| network-production-RT  | rtb-046ac920c29cd79ee        | subnet-04e89e4ee3018b...     | -                 |
| -                      | rtb-072954dc95ffa65cc        | -                            | -                 |

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "IP Subnet" and displays a terminal session on an Amazon Linux 2023 instance. The terminal output includes:

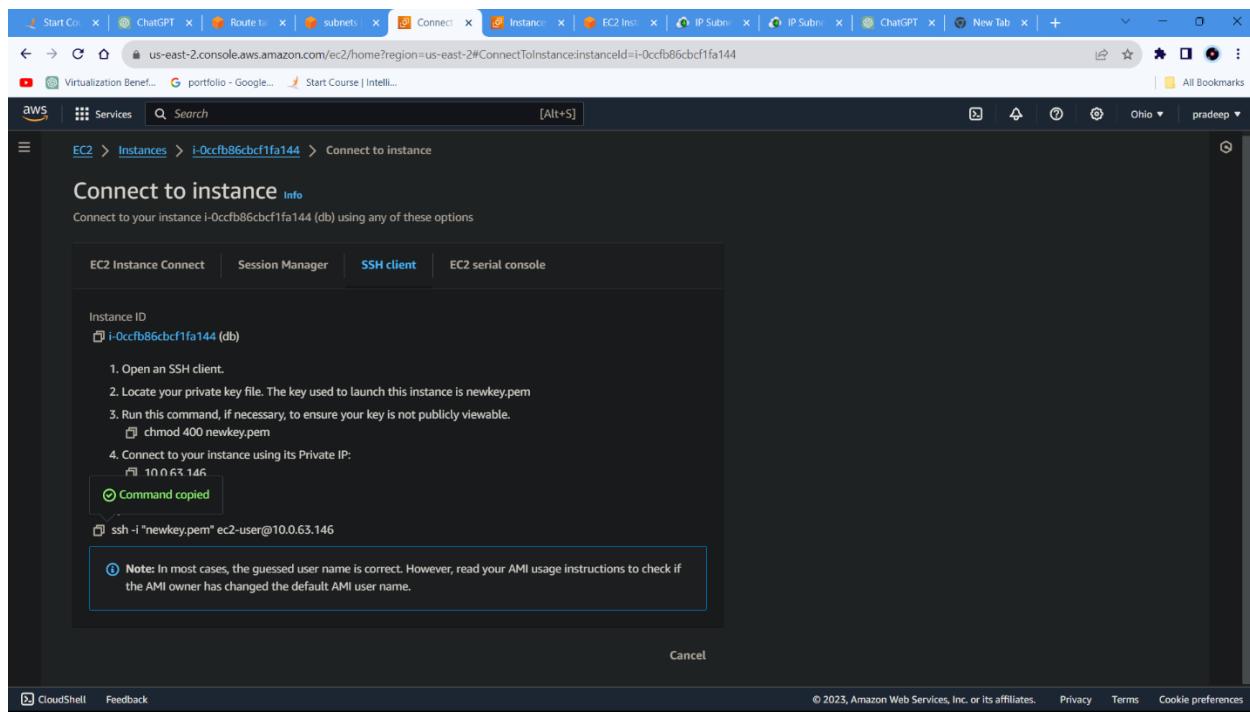
```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Sun Dec 3 10:54:27 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem
```

Below the terminal, the instance details are shown:

i-017cb989e00cf53f4 (web)  
Public IPs: 3.141.152.51 Private IPs: 10.0.73.25

At the bottom, there are CloudShell and Feedback buttons, and a copyright notice: © 2023, Amazon Web Services, Inc. or its affiliates.



A screenshot of an AWS CloudShell terminal window. The title bar shows multiple tabs including "Start Co...", "ChatGPT", "Route t...", "subnets", "Connect", "EC2 Inst...", "IP Subn...", "IP Subn...", "ChatGPT", "New Tab", and "pradeep". The main area displays a terminal session:

```
>Last login: Sun Dec 3 10:54:27 2023 from 49.15.226.14
[ec2-user@ip-10-0-73-23 ~]$ ls
key.pem key.pem.save
[ec2-user@ip-10-0-73-23 ~]$ sudo chmod 400 key.pem
[ec2-user@ip-10-0-73-23 ~]$ sudo ssh -i "key.pem" ec2-user@10.0.63.146
The authenticity of host '10.0.63.146 (10.0.63.146)' can't be established.
ED25519 key fingerprint is SHA256:nzJUR8yoOpt6Q0lKEUqPIvBEkeu/8ZkU8VFc/nvgfGA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.63.146' (ED25519) to the list of known hosts.

[ec2-user@ip-10-0-63-146 ~]$ ping 10.1.18.132
```

The terminal then shows the output of the ping command:

```
i-017cb989e00cf53f4 (web)
Public IPs: 3.141.152.51 Private IPs: 10.0.73.23
```

At the bottom, there are links for "CloudShell" and "Feedback", and a copyright notice: "© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

A second screenshot of an AWS CloudShell terminal window, identical in layout to the first. The terminal session shows the same SSH setup and the execution of a ping command:

```
[ec2-user@ip-10-0-63-146 ~]$ ping 10.1.18.132
```

The terminal then displays the output of the ping command, which includes the number of bytes sent, sequence numbers, and round-trip times:

```
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.
64 bytes from 10.1.18.132: icmp_seq=1 ttl=127 time=0.430 ms
64 bytes from 10.1.18.132: icmp_seq=2 ttl=127 time=0.425 ms
64 bytes from 10.1.18.132: icmp_seq=3 ttl=127 time=0.07 ms
64 bytes from 10.1.18.132: icmp_seq=4 ttl=127 time=0.500 ms
```

The terminal then shows the output of the ping command again:

```
i-017cb989e00cf53f4 (web)
Public IPs: 3.141.152.51 Private IPs: 10.0.73.23
```

At the bottom, there are links for "CloudShell" and "Feedback", and a copyright notice: "© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

```

Start Co... | ChatGPT | Route t... | subnets | Connect | Instance | EC2 Inst... | IP Subn... | IP Subn... | ChatGPT | New Tab | + | - | × |
us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-2&connType=standard&instanceId=i-017cb989e00cf5f3f4&osUser=ec2-user&sshPort=22#/
Virtualization Benef... | portfolio - Google... | Start Course | Intelli...
aws Services Search [Alt+S]
The authenticity of host '10.0.63.146 (10.0.63.146)' can't be established.
ED25519 key fingerprint is SHA256:nzJUR8yoOpt6Q01KEUgPIvB5keu/82ku8VFc/nvgfGA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/(fingerprint))? yes
Warning: Permanently added '10.0.63.146' (ED25519) to the list of known hosts.

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-63-146 ~]$ ping 10.1.18.132
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.
64 bytes from 10.1.18.132: icmp_seq=1 ttl=127 time=0.430 ms
64 bytes from 10.1.18.132: icmp_seq=2 ttl=127 time=0.425 ms
64 bytes from 10.1.18.132: icmp_seq=3 ttl=127 time=1.07 ms
64 bytes from 10.1.18.132: icmp_seq=4 ttl=127 time=0.500 ms
64 bytes from 10.1.18.132: icmp_seq=5 ttl=127 time=0.542 ms
64 bytes from 10.1.18.132: icmp_seq=6 ttl=127 time=0.450 ms
^C
--- 10.1.18.132 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5189ms
rtt min/avg/max/mdev = 0.425/0.570/1.074/0.229 ms
[ec2-user@ip-10-0-63-146 ~]$

i-017cb989e00cf5f3f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23

CloudShell Feedback
© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

```

Start Co... | ChatGPT | Route t... | subnets | Connect | Instance | EC2 Inst... | IP Subn... | IP Subn... | ChatGPT | New Tab | + | - | × |
us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-2&connType=standard&instanceId=i-017cb989e00cf5f3f4&osUser=ec2-user&sshPort=22#/
Virtualization Benef... | portfolio - Google... | Start Course | Intelli...
aws Services Search [Alt+S]
Are you sure you want to continue connecting (yes/no/(fingerprint))? yes
Warning: Permanently added '10.0.63.146' (ED25519) to the list of known hosts.

Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-63-146 ~]$ ping 10.1.18.132
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.
64 bytes from 10.1.18.132: icmp_seq=1 ttl=127 time=0.430 ms
64 bytes from 10.1.18.132: icmp_seq=2 ttl=127 time=0.425 ms
64 bytes from 10.1.18.132: icmp_seq=3 ttl=127 time=1.07 ms
64 bytes from 10.1.18.132: icmp_seq=4 ttl=127 time=0.500 ms
64 bytes from 10.1.18.132: icmp_seq=5 ttl=127 time=0.542 ms
64 bytes from 10.1.18.132: icmp_seq=6 ttl=127 time=0.450 ms
^C
--- 10.1.18.132 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5189ms
rtt min/avg/max/mdev = 0.425/0.570/1.074/0.229 ms
[ec2-user@ip-10-0-63-146 ~]$
[ec2-user@ip-10-0-63-146 ~]$ logout
Connection to 10.0.63.146 closed.
[ec2-user@ip-10-0-73-23 ~]$ ping 10.1.18.132

i-017cb989e00cf5f3f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23

CloudShell Feedback
© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

The screenshot shows a CloudShell terminal window with the following content:

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-63-146 ~]$ ping 10.1.18.132
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.
64 bytes from 10.1.18.132: icmp_seq=1 ttl=127 time=0.430 ms
64 bytes from 10.1.18.132: icmp_seq=2 ttl=127 time=0.425 ms
64 bytes from 10.1.18.132: icmp_seq=3 ttl=127 time=1.07 ms
64 bytes from 10.1.18.132: icmp_seq=4 ttl=127 time=0.500 ms
64 bytes from 10.1.18.132: icmp_seq=5 ttl=127 time=0.542 ms
64 bytes from 10.1.18.132: icmp_seq=6 ttl=127 time=0.450 ms
^C
--- 10.1.18.132 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5189ms
rtt min/avg/max/mdev = 0.425/0.570/1.074/0.229 ms
[ec2-user@ip-10-0-63-146 ~]$ logout
Connection to 10.0.63.146 closed.
[ec2-user@ip-10-0-73-23 ~]$ ping 10.1.18.132
PING 10.1.18.132 (10.1.18.132) 56(84) bytes of data.

i-017cb989e00cf53f4 (web)
PublicIPs: 3.141.152.51 PrivateIPs: 10.0.73.23
```