

Create VPC Infrastructure (Task 1)



Zaeem Attique Ashar

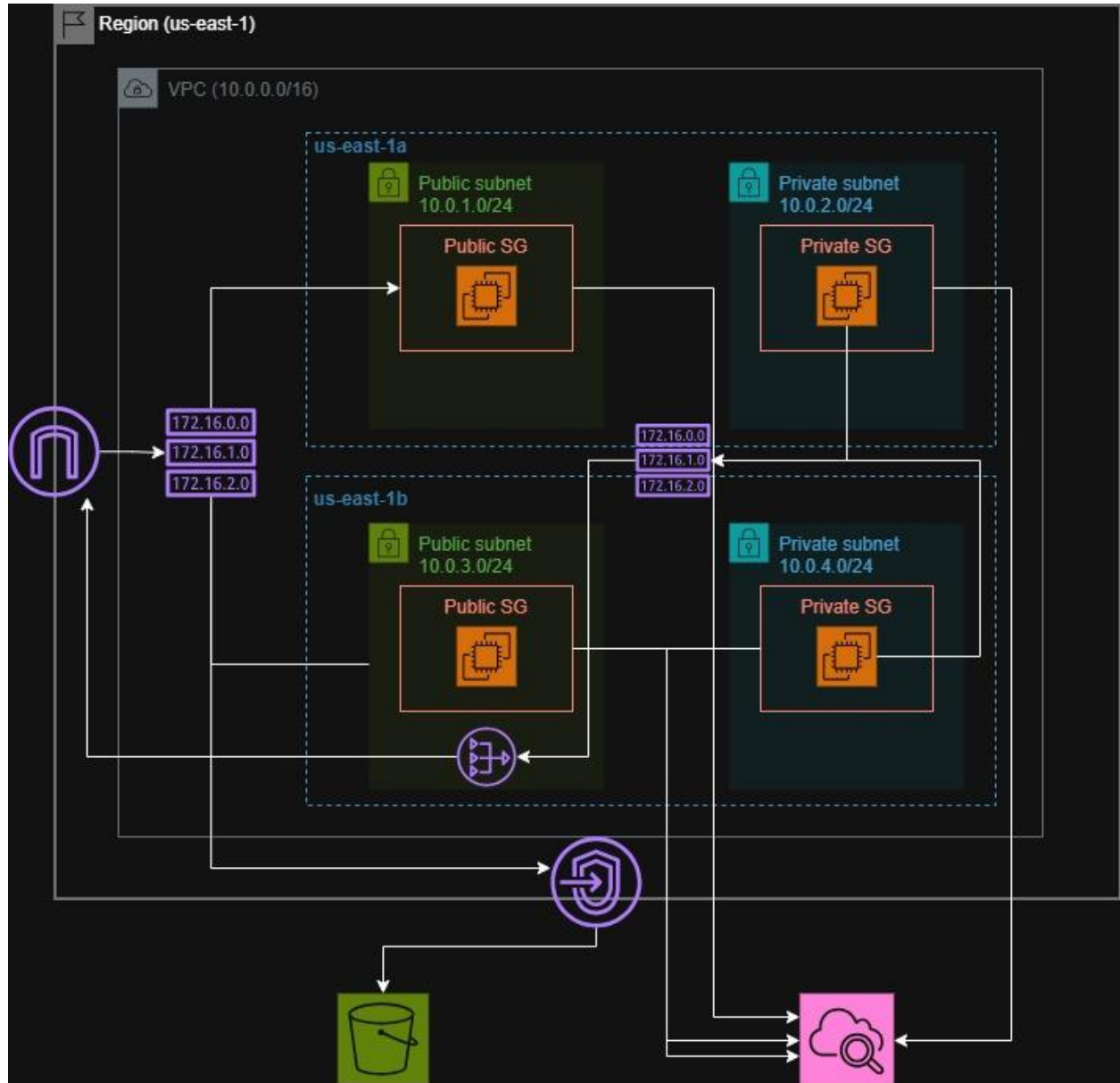
Cloud Intern

Task Description:

Create a custom Virtual Private Cloud (VPC) setup to host secure and scalable AWS resources across public and private subnets with proper routing, NAT, and security configurations. Also enable VPC Flow Logs for network traffic monitoring and store them in S3 Bucket as well by creating VPC Endpoint.

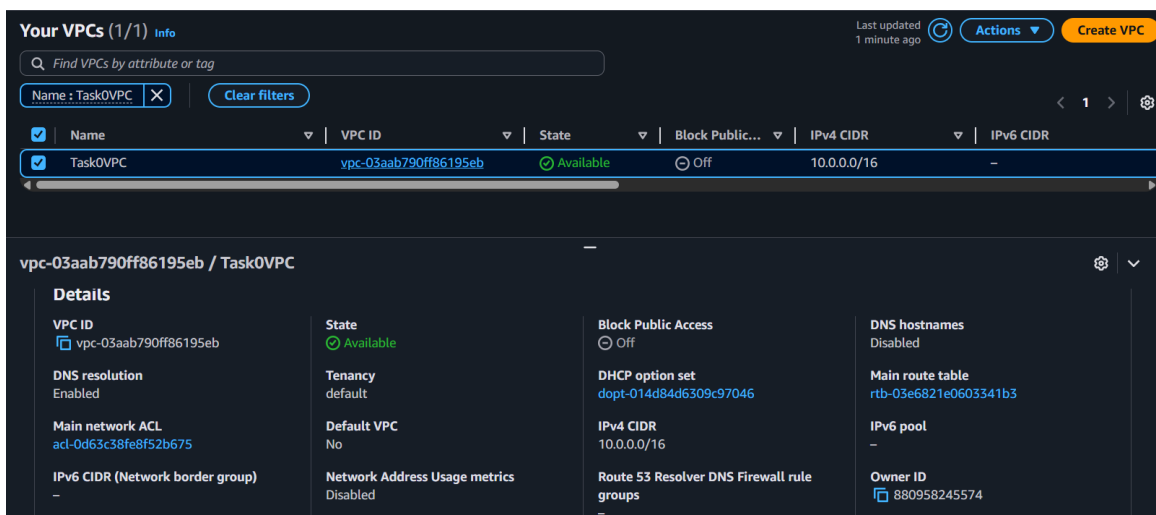
Architecture Diagram:	3
Task 0.1: Create a custom VPC with a specified CIDR block	4
Task 0.2: Define public and private subnets across multiple Availability Zones.....	4
Task 0.3: Set up an Internet Gateway and attach it to the VPC.	5
Task 0.4: Create route tables and associate them with appropriate subnets.	5
Task 0.5: Configure NAT Gateway in a public subnet for private subnet internet access.	7
Task 0.6: Launch EC2 instances in private and public subnets as needed	8
Task 0.7: Implement security groups and network ACLs for traffic control.....	11
Task 0.8: Enable VPC Flow Logs for network traffic monitoring	12
Task 0.9: Create VPC Endpoints for secure private connectivity to AWS services	13
Task 0.10: Use Elastic IPs for NAT Gateway or public-facing resources	14
Task 0.11: Test connectivity between subnets and to the internet	15

Architecture Diagram:



Task 0.1: Create a custom VPC with a specified CIDR block

- Go to the VPC Dashboard and select the Create VPC button.
- Configuration:
 - Resources to create: VPC only
 - Name tag: Task0VPC
 - IPv4 CIDR block: IPv4 CIDR manual input
 - IPv4 CIDR: 10.0.0.0/16
 - IPv6 CIDR block: No IPv6 CIDR block
 - Tenancy: Default
- Click on create VPC.



Task 0.2: Define public and private subnets across multiple Availability Zones

- Go to the subnets tab in the VPC dashboard.
- Click on create subnets button and select the VPC created in the last step.
- Create subnet with the following config:
 - Name: Task0-pub-1a, Zone: us-east-1a, CIDR Block: 10.0.1.0/24
 - Name: Task0-priv-1a, Zone: us-east-1a, CIDR Block: 10.0.2.0/24
 - Name: Task0-pub-1b, Zone: us-east-1b, CIDR Block: 10.0.3.0/24
 - Name: Task0-priv-1b, Zone: us-east-1b, CIDR Block: 10.0.4.0/24

Subnets (4) Info Last updated 1 minute ago Actions Create subnet

Find subnets by attribute or tag

Subnet ID : subnet-0abb6619e9f22d8e7 X Subnet ID : subnet-0a9664e0e2e3d746c X Subnet ID : subnet-09c710bbff34ae0bc X Show more (+1)

Clear filters

<input type="checkbox"/>	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
<input type="checkbox"/>	Task0-private-sn-1a	subnet-0a9664e0e2e3d746c	Available	vpc-03aab790ff86195eb Task...	Off	10.0.2.0/24
<input type="checkbox"/>	Task0-private-sn-1b	subnet-09124e18a3f5c5a9b	Available	vpc-03aab790ff86195eb Task...	Off	10.0.4.0/24
<input type="checkbox"/>	Task0-public-sn-1a	subnet-0abb6619e9f22d8e7	Available	vpc-03aab790ff86195eb Task...	Off	10.0.1.0/24
<input type="checkbox"/>	Task0-public-sn-1b	subnet-09c710bbff34ae0bc	Available	vpc-03aab790ff86195eb Task...	Off	10.0.3.0/24

Task 0.3: Set up an Internet Gateway and attach it to the VPC.

- Go to the Internet Gateways tab in the VPC Dashboard.
- Use the Create VPC button.
- Name your VPC and click Create.
- Now click on the Actions button and choose to attach VPC.
- Select the VPC previously created for the task and click on the attach IGW button.

igw-05d011edb29557134 / Task0-igw Actions

Details <small>Info</small>	State	VPC ID	Owner
Internet gateway ID igw-05d011edb29557134	Attached	vpc-03aab790ff86195eb Task0VPC	880958245574

Tags (1) Manage tags

Key	Value
Name	Task0-igw

Task 0.4: Create route tables and associate them with appropriate subnets.

- Go to route tables tab on the VPC Dashboard.
- Click on create route table and name the public route table.
- Select the VPC created for this task.
- Hit create route table.
- Repeat above steps to create a private route table.
- Now associate the public RT with the public subnet and private RT with private subnets.

Route tables (1/3) Info Last updated 1 minute ago Actions Create route table

Find route tables by attribute or tag

task0 Clear filters

	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input type="checkbox"/>	Task0-public-rt	rtb-01d2102b6c7e69894	2 subnets	-	No	vpc-03aab790ff86195eb Tasi
<input checked="" type="checkbox"/>	Task0-private-rt	rtb-0025f22ff12bf885e	2 subnets	-	No	vpc-03aab790ff86195eb Tasi

rtb-0025f22ff12bf885e / Task0-private-rt

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

Explicit subnet associations (2) Edit subnet associations

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Task0-private-sn-1a	subnet-0a9664e0e2e3d746c	10.0.2.0/24	-
Task0-private-sn-1b	subnet-09124e18a3f5c5a9b	10.0.4.0/24	-

Route tables (1/3) Info Last updated 2 minutes ago Actions Create route table

Find route tables by attribute or tag

task0 Clear filters

	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input checked="" type="checkbox"/>	Task0-public-rt	rtb-01d2102b6c7e69894	2 subnets	-	No	vpc-03aab790ff86195eb Tasi
<input type="checkbox"/>	Task0-private-rt	rtb-0025f22ff12bf885e	2 subnets	-	No	vpc-03aab790ff86195eb Tasi

rtb-01d2102b6c7e69894 / Task0-public-rt

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

Explicit subnet associations (2) Edit subnet associations

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Task0-public-sn-1a	subnet-0abb6619e9f22d8e7	10.0.1.0/24	-
Task0-public-sn-1b	subnet-09c710bbff34ae0bc	10.0.3.0/24	-

- Go to the public route table and create the following route(s)
 - Instance to IGW:
 - Destination: 0.0.0.0/0
 - Target: igw

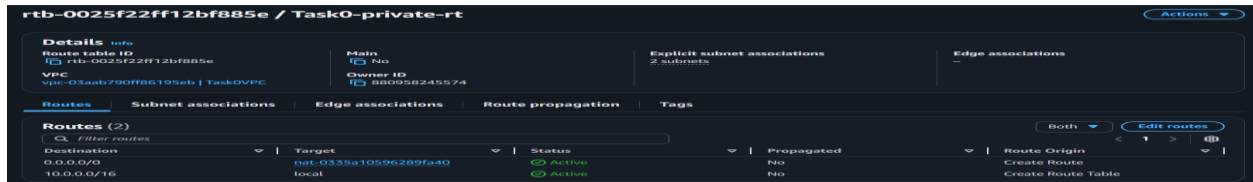
Routes (2) Both Edit routes

Filter routes

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-05d011edb29557134	Active	No	Create Route
10.0.0.0/16	local	Active	No	Create Route Table

- Go to the private route table and create the following route
 - Instance to NGW:
 - Destination: 0.0.0.0/0

- Target: NGW



Task 0.5: Configure NAT Gateway in a public subnet for private subnet internet access.

- Go to the NAT Gateway tab in the VPC Dashboard.
- Click on create NAT Gateway, set a name for the NAT Gateway.
- Select the public subnet Task0-public-sn-1a.
- Click on the allocate elastic IP button and then create the NAT Gateway.



- Follow the same steps to create another NAT Gateway in the Task0-private-sn-1b.

nat-0335a10596289fa40 / Task0-ngw-1a Actions ▾

Details

NAT gateway ID
nat-0335a10596289fa40

NAT gateway ARN
arn:aws:ec2:us-east-1:880958245574:natgateway/nat-0335a10596289fa40

VPC
vpc-03aab790ff86195eb / Task0VPC

Connectivity type
Public

Primary public IPv4 address
-

Subnet
subnet-09c710bbff34ae0bc / Task0-public-sn-1b

State
Pending

Primary private IPv4 address
-

Created
Wednesday, November 5, 2025 at 19:27:38 GMT+5

State message [Info](#)
-

Primary network interface ID
-

Deleted
-

[Secondary IPv4 addresses](#) | [Monitoring](#) | [Tags](#)

Secondary IPv4 addresses 🔄 [Edit secondary IPv4 address associations](#)

🔍 Search

Private IPv4 address	Network interface ID	Status	Failure message
Secondary IPv4 addresses are not available for this nat gateway.			

Task 0.6: Launch EC2 instances in private and public subnets as needed

- Go to the EC2 Dashboard and click on the launch instance button.
- Configure the Public instance 1 as follows:
 - Name: Task0-public-1a
 - AMI: Amazon Linux 2023 (x86)
 - Instance type: t3.micro
 - Keypair: Proceed without key pair
 - VPC: Task0VPC
 - Subnet: Task0-private-sn-1a
 - Auto-assign public IP: Enabled
 - Create security group, rule: Type=HTTP, Source Type=Anywhere
 - Storage: 8GiB, gp3
 - Connect via SSH and run command *sudo dnf install nginx*

Instance summary for i-0dbb28dee54c1b9ae (Task0-Public-1a) [Info](#) 🔄 [Connect](#) [Instance state ▾](#) [Actions ▾](#)

Updated 10 minutes ago

Instance ID i-0dbb28dee54c1b9ae	Public IPv4 address 3.95.165.154 open address	Private IPv4 addresses 10.0.1.62
IPv6 address -	Instance state Running	Public DNS -
Hostname type IP name: ip-10-0-1-62.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-1-62.ec2.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t3.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 3.95.165.154 [Public IP]	VPC ID vpc-03aab790ff86195eb (Task0VPC)	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-0abb6619e9f22d8e7 (Task0-public-sn-1a)	Managed false
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:880958245574:instance/i-0dbb28dee54c1b9ae	
Operator -		

- Configure the Public instance 2 as follows:
 - Name: Task0-public-1b
 - AMI: Amazon Linux 2023 (x86)
 - Instance type: t3.micro
 - Keypair: Proceed without key pair
 - VPC: Task0VPC
 - Subnet: Task0-private-sn-1b
 - Auto-assign public IP: Enabled
 - Create security group, rule: Type=HTTP, Source Type=Anywhere
 - Storage: 8GiB, gp3
 - Connect via SSH and run command `sudo dnf install nginx`

Instance summary for i-01a81a23b5d1572ad (Task0-public-1b) Info

Updated less than a minute ago

Instance ID i-01a81a23b5d1572ad	Public IPv4 address 54.226.58.189 open address	Private IPv4 addresses 10.0.3.181
IPv6 address -	Instance state Running	Public DNS -
Hostname type IP name: ip-10-0-3-181.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-3-181.ec2.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t3.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 54.226.58.189 [Public IP]	VPC ID vpc-03aab790ff86195eb (Task0VPC)	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-09c710bbff34ae0bc (Task0-private-sn-1b)	Managed false
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:880958245574:instance/i-01a81a23b5d1572ad	
Operator -		

- Configure the Private instance 1 as follows:
 - Name: Task0-private-1a
 - AMI: Amazon Linux 2023 (x86)
 - Instance type: t3.micro
 - Keypair: Proceed without key pair
 - VPC: Task0VPC
 - Subnet: Task0-private-sn-1a
 - Auto-assign public IP: Enabled
 - Create security group, rule: Type=HTTP, Source Type=Anywhere
 - Attach Role: ssmManagedInstanceCore
 - Storage: 8GiB, gp3

Instance summary for i-0fff2951751421d35 (Task0-private-1a) Info

Updated 2 minutes ago

[Refresh](#)
[Connect](#)
[Instance state ▼](#)
[Actions ▼](#)

Instance ID i-0fff2951751421d35	Public IPv4 address -	Private IPv4 addresses 10.0.2.109
IPv6 address -	Instance state Running	Public DNS -
Hostname type IP name: ip-10-0-2-109.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-2-109.ec2.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t3.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address -	VPC ID vpc-03aab790ff86195eb (Task0VPC) ↗	Auto Scaling Group name -
IAM Role role-for-ssm-instance-core ↗	Subnet ID subnet-0a9664e0e2e3d746c (Task0-private-sn-1a) ↗	Managed false
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:880958245574:instance/i-0fff2951751421d35	
Operator -		

- Configure the Private instance 2 as follows:
 - Name: Task0-private-1b
 - AMI: Amazon Linux 2023 (x86)
 - Instance type: t3.micro
 - Keypair: Proceed without key pair
 - VPC: Task0VPC
 - Subnet: Task0-private-sn-1b
 - Auto-assign public IP: Enabled
 - Attach Role: ssmManagedInstanceCore
 - Create security group, rule: Type=HTTP, Source Type=Anywhere
 - Storage: 8GiB, gp3

Instance summary for i-02734a617f23182e7 (Task0-private-1b) Info

Updated less than a minute ago

[Refresh](#)
[Connect](#)
[Instance state ▼](#)
[Actions ▼](#)

Instance ID i-02734a617f23182e7	Public IPv4 address -	Private IPv4 addresses 10.0.4.77
IPv6 address -	Instance state Running	Public DNS -
Hostname type IP name: ip-10-0-4-77.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-4-77.ec2.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t3.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address -	VPC ID vpc-03aab790ff86195eb (Task0VPC) ↗	Auto Scaling Group name -
IAM Role role-for-ssm-instance-core ↗	Subnet ID subnet-09124e18a3f5c5a9b (Task0-private-sn-1b) ↗	Managed false
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:880958245574:instance/i-02734a617f23182e7	
Operator -		

Task 0.7: Implement security groups and network ACLs for traffic control

Security Group for public instance in SN 1a:

- In the EC2 Dashboard, go to the security groups tab.
- Click on the create security group button.
- Give the new SG a name, Select the VPC: Task0VPC
- Now create the following inbound rules.
 - Type: HTTP, Source: 0.0.0.0/0 (Hosting Web Server)
 - Type: HTTPS, Source: 0.0.0.0/0 (Secure connection to the webserver)
 - Type: ICMP, Source: 0.0.0.0/0 (Test traffic like ping)
 - Type: SSH, Source: 0.0.0.0/0 (Connecting to the machine via SSH)

Create the same SG for the public instance in SN 1b.

The screenshot displays the AWS Management Console interface for a security group. The top section shows the security group's details, including its name, ID, description, VPC ID, owner, and rule counts. Below this, the 'Inbound rules' tab is selected, showing a table of four inbound rules: HTTP (port 80), HTTPS (port 443), All ICMP - IPv4 (port All), and SSH (port 22). The rules are all configured for IPv4 and allow traffic from 0.0.0.0/0.

sg-0ef892bb8957b3146 - launch-wizard-4						
Details						
Security group name launch-wizard-4	Security group ID sg-0ef892bb8957b3146	Description launch-wizard-4 created 2025-11-05T 15:28:52.561Z	VPC ID vpc-03aab790ff86195eb	Owner 880958245574	Inbound rules count 4 Permission entries	Outbound rules count 1 Permission entry
Inbound rules (4)						
Name	Security group rule ID	IP version	Type	Protocol	Port range	
-	sgr-0161d9e8e0d0252be	IPv4	HTTP	TCP	80	
-	sgr-01a9de096a057c946	IPv4	HTTPS	TCP	443	
-	sgr-0659b9fbb214ae49d	IPv4	All ICMP - IPv4	ICMP	All	
-	sgr-00436f9f212befeb2	IPv4	SSH	TCP	22	

Security Group for private instance in SN 1a:

- In the EC2 Dashboard, go to the security groups tab.
- Click on the create security group button.
- Give the new SG a name, Select the VPC: Task0VPC
- Now create the following inbound rules.
 - Type: HTTP, Source: 0.0.0.0/0 (for connecting via SSM)
 - Type: All Traffic, Destination: 0.0.0.0/0 (sending traffic to NGW)

Create the same SG for the private instance in SN 1b

sg-01822ec0b0fbcdb8 - launch-wizard-6 Actions ▾

Details

Security group name
launch-wizard-6

Owner
880958245574

Security group ID
sg-01822ec0b0fbcdb8

Inbound rules count
2 Permission entries

Description
launch-wizard-6 created 2025-11-05T16:00:52.288Z

Outbound rules count
1 Permission entry

VPC ID
vpc-03aab790ff86195eb [🔗](#)

Inbound rules | Outbound rules | Sharing - new | VPC associations - new | Tags

Inbound rules (2) Manage tags Edit inbound rules

Search

<input type="checkbox"/>	Name ▾	Security group rule ID ▾	IP version ▾	Type ▾	Protocol ▾	Port range ▾
<input type="checkbox"/>	-	sgr-0e281a5dad4cb731d	IPv4	HTTP	TCP	80
<input type="checkbox"/>	-	sgr-0cbbdcd11143d36e6	IPv4	SSH	TCP	22

Task 0.8: Enable VPC Flow Logs for network traffic monitoring

- Go to CloudWatch Dashboard
- Go to logs -> log groups tab.
- Create a new log group, give it the name Task0-VPC-Flow.
- Now go to the VPC Dashboard and open your VPC tab.
- Select VPC Task0VPC from the list.
- Now use the action dropdown and select create flow log.
- Using the following configurations:
 - Name: Task0-FlowLogs
 - Filter: all
 - Maximum aggregation interval: 10 min
 - Destination: Save to CW Logs
 - Destination log group: Task0-VPC-Flow
 - Service access: Create and use a new service role
 - Log record format: AWS default format
- Click on the create Flow Log button.

Details			
Flow Log ID fl-0df8854182cce9574	Destination Type cloud-watch-logs	Traffic Type All	File Format -
Name Task0-FlowLogs	Destination Name Task0-VPC-Flow	Max Aggregation Interval 10 minutes	Hive Compatible Partitions -
State Active	IAM Role arn:aws:iam::880958245574:role/service-role/VPCLowLogs-Cloudwatch-1762362047093	Log Format Default	Partition Logs -
Creation Time Wednesday, November 5, 2025 at 22:06:21 GMT+5	Cross Account IAM Role -		

- Create an S3 bucket for demonstration purposes.
- Go to the VPC Dashboard and open the Endpoints tab.
- Following is the configuration:
 - Name: task0-s3-endpoint
 - Type: AWS Services
 - Service Name: com.amazonaws.us-east-1.s3 (Gateway)
 - VPC: Task0VPC
 - Route Table: Task0-rt-public
 - Policy: Full Access

Task 0.10: Use Elastic IPs for NAT Gateway or public-facing resources

- While creating the NGW, use the allocate elastic IP button to auto-assign EIP.

The screenshot shows the AWS Elastic IP addresses console. At the top, there's a search bar and a filter for 'Public IPv4 address: 34.204.216.110'. Below the search bar, a table lists the Elastic IP addresses. The first entry is selected, showing details for 34.204.216.110. The details panel on the right shows the following information:

Summary	Type	Allocation ID	Reverse DNS record
Allocated IPv4 address 34.204.216.110	Public IP	eipalloc-0ec8a00cce80f29c6	-
Association ID eipassoc-0bd4151767d363ebf	Scope VPC	Associated instance ID -	Private IP address 10.0.3.28
Network interface ID eni-044fb2c0e450c3f04	Network interface owner account ID 880958245574	Public DNS -	NAT Gateway ID nat-0335a10596289fa40 (Task0-ngw-1b)
Address pool Amazon	Network border group us-east-1	Service managed -	

- Now go to the VPC Dashboard and go to the Elastic IP tab.
- Select the allocate elastic IP button. Continue with the following configuration.
 - Amazon's pool of IPv4 addresses
 - Network Border Group: us-east-1.
- Then create the EIP.
- From the EIP dashboard, go to actions and attach the EIP to a public facing EC2 Instance ().

The screenshot shows the AWS Elastic IP addresses console for a different address, 44.221.133.223. The details panel on the right shows the following information:

Summary	Type	Allocation ID	Reverse DNS record
Allocated IPv4 address 44.221.133.223	Public IP	eipalloc-0c4311eb77d4696ff	-
Association ID eipassoc-0a20fd2f973f2529a	Scope VPC	Associated instance ID i-0d8bb28dee34c1b9aa	Private IP address 10.0.1.62
Network interface ID eni-0a17fbc5e31d0650	Network interface owner account ID 880958245574	Public DNS -	NAT Gateway ID -
Address pool Amazon	Network border group us-east-1	Service managed -	

Below the details panel, there's a 'Tags(0)' section with a 'Manage tags' button. The message states: 'No tags associated with this resource. Click the Manage tags button to add your first tag.'

- 98.90.117.196

Actions Associate Elastic IP address

Summary

Allocated IPv4 address

98.90.117.196

Association ID

eipassoc-09a664d89fb6da4d9

Network interface ID

eni-026e62af3a9430e06

Address pool

Amazon

Type

Public IP

Scope

VPC

Network interface owner account ID

880958245574

Network border group

us-east-1

Allocation ID

eipalloc-0a6f18d3f31dbe89f

Associated instance ID

i-01a81a23b5d1572ad

Public DNS

Service managed

Reverse DNS record

Private IP address

10.0.3.181

NAT Gateway ID

Tags(0)

Manage tags

Key

Value

No tags associated with this resource

Click the Manage tags button to add your first tag

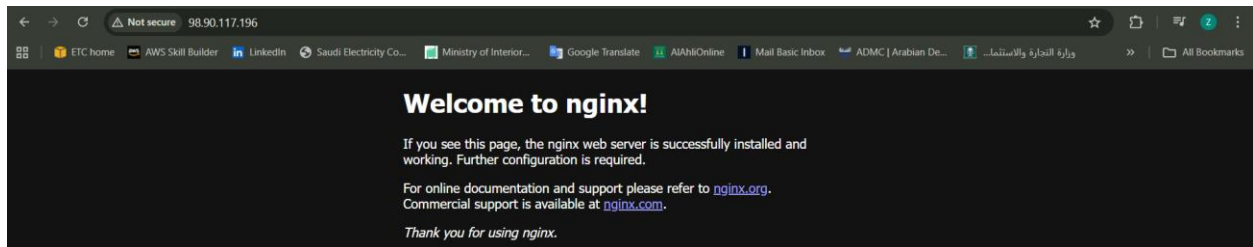
Manage tags

Task 0.11: Test connectivity between subnets and to the internet

- ```

~\#### Amazon Linux 2023
~~\#####
~~\###|
~~\#/
~~~~v-! ->  
  
~~~~  
~~~~.  
~~~~/_/!  
Last login: Wed Nov 5 16:12:18 2025 from 18.206.107.29
[ec2-user@ip-10-0-3-181 ~]$ ping www.google.com
PING www.google.com (142.251.111.105) 56(84) bytes of data:
64 bytes from bk-in-f105.1e100.net (142.251.111.105): icmp_seq=1 ttl=106 time=1.67 ms
64 bytes from bk-in-f105.1e100.net (142.251.111.105): icmp_seq=2 ttl=106 time=1.73 ms
64 bytes from bk-in-f105.1e100.net (142.251.111.105): icmp_seq=3 ttl=106 time=1.72 ms
^C
--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.671/1.707/1.732/0.026 ms
[ec2-user@ip-10-0-3-181 ~]$
```
- i-01a81a23b5d1572ad (Task0-public-1b)
- PublicIPs: 98.90.117.196 PrivateIPs: 10.0.3.181

- Enter the EIP of public machine to test if HTTP traffic flows via IGW and is allowed by the SG:



The same can be done with the other public EC2 instance to check connectivity.

- Connect to the private EC2 instance using the Session Manager via NGW
- Use the ping command to check if the traffic is flowing through the NGW.

Session ID: Zaccm-6i5azgyj4ltit8fblp6lq85h8

Shortcuts

Instance ID: i-02734a617f23182c7

```
sh-5.2$ ping www.google.com
PING www.google.com (142.250.31.104) 56(84) bytes of data.
64 bytes from bj-in-f104.1e100.net (142.250.31.104): icmp_seq=1 ttl=105 time=2.38 ms
64 bytes from bj-in-f104.1e100.net (142.250.31.104): icmp_seq=2 ttl=105 time=1.57 ms
64 bytes from bj-in-f104.1e100.net (142.250.31.104): icmp_seq=3 ttl=105 time=1.54 ms
64 bytes from bj-in-f104.1e100.net (142.250.31.104): icmp_seq=4 ttl=105 time=1.60 ms
^C
--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.544/1.772/2.377/0.349 ms
sh-5.2$
```

- To check the flow logs, head over to the CloudWatch Dashboard.
- Click on logs and then open the logs group tab.
- Select an interface to check the logs, and they will be displayed.

