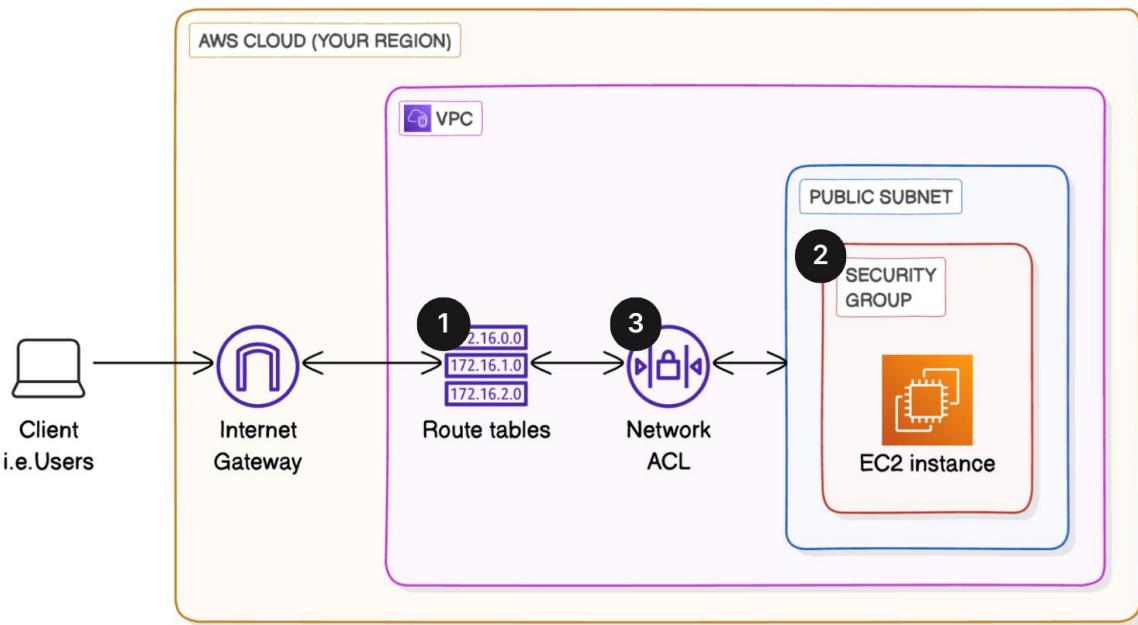


VPC TRAFFIC FLOW AND SECURITY - STEP-BY-STEP DOCUMENTATION



This documentation provides a comprehensive, step-by-step process for setting up and securing an Amazon VPC. It builds on the earlier project 'Build a Virtual Private Cloud' and includes both the foundational steps and the new steps for configuring traffic flow and security.

Step 1: Create a VPC

Set up a new Virtual Private Cloud in AWS.

Actions:

- Log in to AWS Management Console.
- Search for 'VPC' in the search bar and select it.
- In the left-hand navigation pane, choose 'Your VPCs'.
- Click 'Create VPC'.
- Select 'VPC only' option.
- Enter a Name tag (e.g., NextWork VPC).
- Enter an IPv4 CIDR block (e.g., 10.0.0.0/16).
- Click 'Create VPC' to finalize.

Your VPCs (1/2) Info

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
vpc-0dcc2a2993f222607	Available	Off	172.31.0.0/16	-	-
Nextwork-VPC	Available	Off	10.0.0.0/16	-	-

Details

VPC ID vpc-04f45aa199fe73f17	State Available	Block Public Access Off	DNS hostnames Disabled
DNS resolution Enabled	Tenancy default	DHCP option set dopt-019ae45123624041f	Main route table -
Main network ACL -	Default VPC No	IPv4 CIDR 10.0.0.0/16	IPv6 pool -
IPv6 CIDR (Network border group) -	Network Address Usage metrics Disabled	Route 53 Resolver DNS Firewall rule groups -	Owner ID 944362433406

Step 2: Create Subnets

Divide your VPC into subnets for better resource organization.

Actions:

- From the VPC Dashboard, select 'Subnets' in the navigation pane.
- Click 'Create subnet'.
- Choose the VPC you just created (NextWork VPC).
- Enter a Subnet name (e.g., Public 1).
- Select an Availability Zone.
- Enter an IPv4 CIDR block for the subnet (e.g., 10.0.0.0/24).
- Click 'Create subnet'.
- Select your new subnet and go to 'Edit subnet settings'.
- Enable 'Auto-assign public IPv4 address'.
- Click 'Save' to apply changes.

subnet-0f25d9505300126c9 / Public-01

Details

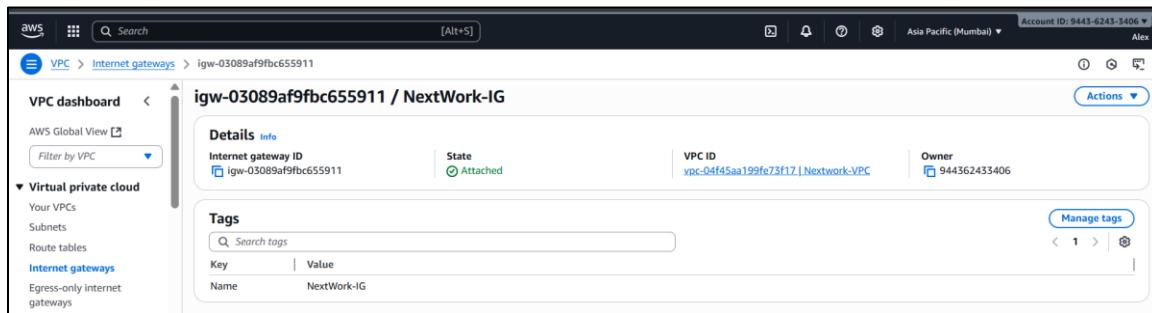
Subnet ID subnet-0f25d9505300126c9	Subnet ARN arn:aws:ec2:ap-south-1:944362433406:subnet/subnet-0f25d9505300126c9	State Available	Block Public Access Off
IPv4 CIDR 10.0.0.0/24	Available IPv4 addresses 251	IPv6 CIDR -	IPv6 CIDR association ID -
Availability Zone ap-s1:az1 (ap-south-1a)	Network border group ap-south-1	VPC vpc-04f45aa199fe73f17 Nextwork-VPC	Route table rtb-00ca3570786dcbdb82
Network ACL acl-0872c7e9b4729c3af	Default subnet No	Auto-assign public IPv4 address Yes	Auto-assign IPv6 address No
Auto-assign customer-owned IPv4 address No	Customer-owned IPv4 pool -	Outpost ID -	IPv4 CIDR reservations -
IPv6 CIDR reservations -		Hostname type IP	Resource name DNS A record -

Step 3: Create an Internet Gateway

Attach an internet gateway to your VPC to enable internet access.

Actions:

- In the VPC Dashboard, select 'Internet gateways' from the left-hand panel.
- Click 'Create internet gateway'.
- Enter a Name tag (e.g., NextWork IG).
- Click 'Create internet gateway'.
- Select your newly created gateway and choose 'Attach to VPC'.
- Select your VPC (NextWork VPC) and confirm.
- Your VPC is now connected to the internet.



Step 4: Create a Route Table

Set up a route table to define how traffic flows in your VPC.

Actions:

- In the VPC Dashboard, select 'Route tables' from the navigation pane.
- Click 'Create route table'.
- Enter a Name tag (e.g., NextWork-RouteTable).
- Choose your VPC (NextWork VPC).
- Click 'Create route table'.
- Select your new route table and go to the 'Routes' tab.
- Click 'Edit routes' and add a route with Destination = 0.0.0.0/0 and Target = your Internet Gateway.
- Click 'Save changes'.
- Associate the route table with your public subnet by going to the 'Subnet associations' tab and selecting 'Edit subnet associations'.
- Choose your Public 1 subnet and save.

The screenshot shows the AWS VPC Route tables page. The route table ID is 'rtb-0c21e8908f60469f5'. It has no main entry and is owned by 'vpc-04f45aa199fe73f17 | Nextwork-VPC'. Under 'Explicit subnet associations', it lists a single association for 'Public-01' with CIDR '10.0.0.0/24' pointing to subnet 'subnet-0f25d9505300126c9'. The 'Actions' button is visible at the top right.

Step 5: Create a Security Group

Create a security group to control inbound and outbound traffic for your resources.

Actions:

- In the VPC Dashboard, select 'Security groups'.
- Click 'Create security group'.
- Enter a Name tag (e.g., NextWork-SG).
- Select your VPC (NextWork VPC).
- Add inbound rules (e.g., allow HTTP on port 80, SSH on port 22).
- Add outbound rules (default allows all traffic).
- Click 'Create security group' to save.

The screenshot shows the AWS Security Groups page. The security group name is 'Nextwork-SecurityG'. It has a security group ID 'sg-027b505428d88b976', owner '944362433406', and two permission entries. Under 'Inbound rules', there are two rules: one for port 80 (HTTP) and one for port 22 (SSH). The 'Actions' button is visible at the top right.

Step 6: Create a Network ACL (Access Control List)

Set up a network ACL to provide an additional layer of security at the subnet level.

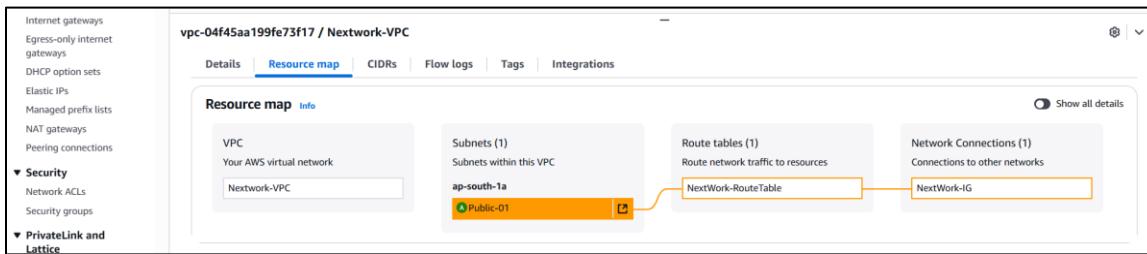
Actions:

- In the VPC Dashboard, select 'Network ACLs' from the navigation pane.
- Click 'Create network ACL'.
- Enter a Name tag (e.g., NextWork-NACL).
- Choose your VPC (NextWork VPC).
- Click 'Create network ACL'.

- Select your new NACL and go to the 'Inbound rules' tab. Add rules (e.g., allow HTTP, HTTPS, and SSH).
- Go to the 'Outbound rules' tab and configure rules (e.g., allow all traffic).
- Associate your NACL with the public subnet by selecting 'Subnet associations'.
- Click 'Edit subnet associations', select your Public 1 subnet, and save.

The screenshot shows the AWS VPC Network ACLs console. On the left, there's a sidebar with various VPC-related services like Subnets, Route tables, Internet gateways, etc. The main area shows a list of Network ACLs with columns for Name, Network ACL ID, Associated with, Default, and VPC ID. One entry is selected: 'NextWork-Network-... acl-0872c7e9b4729c3af subnet-0f25d9505300126c9 / Public-01'. Below this, a detailed view for 'acl-0872c7e9b4729c3af / NextWork-Network-NACL' is shown with tabs for Details, Inbound rules, Outbound rules, Subnet associations, and Tags. The Details tab shows the Network ACL ID, Associated with (subnet-0f25d9505300126c9 / Public-01), Owner (944362433406), Default (Yes), and VPC ID (vpc-04f45aa199fe73f17 / Nextwork-VPC).

VPC Resource Map -



- Client/User:** A user enters the URL of your website into their web browser and hits enter.
- Internet Gateway:** The request is sent from the user's browser through the internet and reaches your internet gateway, **NextWork IG**.
- VPC:** The internet gateway forwards the user's request to the VPC it's attached to, **NextWork VPC**.
- Route Table:** Your VPC has a route table for your public subnet (called **NextWork route table**), which directs traffic to your EC2 instance hosting the website. The user's request gets put on the **local** route in the route table.
- Network ACL:** While en route to your EC2 instance, the request has to pass through the network ACL associated with your public subnet. The network ACL has an inbound rule (rule 100) that lets in traffic from anywhere (0.0.0.0/0), so your request is let through.
- Public Subnet:** The request enters your public subnet **Public 1** and travels to your EC2 instance within the subnet.

7.  **Security Group:** The request reaches the security group **NextWork Security Group** attached to the EC2 instance. The security group has an inbound rule that allows HTTP traffic (Port 80) from anywhere (0.0.0.0/0), so the request can pass through.
8.  **EC2 Instance:** The request reaches your EC2 instance hosting the website. The web server on the EC2 instance processes the request and prepares the response.
9.  **Data gets sent back:** Website content is sent back to the user. The outbound traffic goes through the security group, public subnet, network ACL, route table, VPC, and internet gateway, and user gets to see website content load on their page.