Set up your VPC basics

We're repeating our steps from the first networking project to set up our VPC, subnet and internet gateway. Let's go!

Create a VPC:

Off we go! Your VPC is the foundation for the rest of this project, and represents your corner of the AWS Cloud.

- Log in to your AWS Account..
- In your **AWS Management Console's** search bar, search for VPC.
- Select **VPC** from the drop down menu.
- In the left navigation pane, choose **Your VPCs**.

- Make sure you're on the Region that's closest to you. Use the dropdown on the top right hand corner to switch Regions.
- Choose **Create VPC**.
- Choose **VPC Only**.
- Name tag: NextWork VPC
- IPv4 CIDR: 10.0.0.0/16
- Select **Create VPC**.

Create Subnets:

Nice! We've created our VPC, so it's time for the next step...creating a public subnet.

Quick recap: Subnets are subdivisions within your VPC where you can launch AWS resources.

- In the **VPC Dashboard**, under **Virtual Private Cloud**, choose **Subnets**.
- Choose **Create subnet**.
- Configure your subnet settings:
    - VPC ID: NextWork VPC
    - Subnet name: Public 1
    - Availability Zone: **Select the first Availability Zone in the list.**
    - IPv4 VPC CIDR block: 10.0.0.0/16
    - IPv4 subnet CIDR block: 10.0.0.0/24

  💡 **Is my Public 1 subnet a public subnet?**
  Even though your subnet is labeled **Public 1**, it isn't a public subnet yet. A public subnet must have a route to an **internet gateway**, which you'll attach in a minute.

- Choose **Create subnet.**

- Select the checkbox next to **Public 1**.
- In the **Actions** menu, select **Edit subnet settings.**
- Check the box next to **Enable auto-assign public IPv4 address.**

  💡 **What does it mean to enable auto-assign public IPv4 address?**
  When you enable auto-assign public IPv4 address for a subnet, any **EC2 instance** launched in that subnet will automatically receive a public IP address. This makes the instance accessible from the internet without needing to manually assign a public IP - a huge time saver!

- Choose **Save.**

Create an internet gateway:

Time to connect our VPC to the internet... with an internet gateway:

- In the left navigation pane, choose **Internet gateways**.
- Choose **Create internet gateway**.
- Configure your internet gateway settings:
  - **Name tag:** NextWork IG
- Choose **Create internet gateway**.
- Select your newly created internet gateway and choose **Actions**, then **Attach to VPC**.
- Select **NextWork VPC**.
- Select **Attach internet gateway**.

  💡 **What does attaching an internet gateway to a VPC mean?**
  Attaching an internet gateway means resources in your VPC can now access the internet. The EC2 instances with public IP addresses become accessible to users, so any application you host on those instances become public too.

Nice!

Set up work is allll done... let's dive into what's next for your VPC. 🤿

Create a route table

Even though you've created an internet gateway and attached it to your VPC, you still have to tell the resource in your public subnet how to get to the internet.

You'll have to set up **route tables** to direct traffic from your resource to your internet gateway!

- In the left navigation pane, choose **Route tables**.

  💡 **What is a route table?**
  Think of a **route table** as a GPS for the resources in your subnet. Just like a GPS

helps people get to their destination in a city, a route table is a table of rules, called **routes**, that decide where the data in your network should go.

Every subnet in your VPC needs to be linked to a route table, because the table tells your subnet's traffic where to travel to send and receive data. For example, if you have a web server (i.e. an EC2 instance) hosting a website, the EC2 instance's subnet needs a route table that knows how to direct incoming traffic to the website.

💡 **What's the link between internet gateways and route tables?**
When a subnet's route table has a route that directs internet-bound traffic to the internet gateway, the subnet becomes a **public** subnet. This means your subnet can communicate with the internet.

- Refresh your page.
- Ooo, two route tables! Why are there two?
    - Note: If you see more than two route tables, those route tables would've been set up in other projects that you've completed. Check the **VPC** column in the far right side of the table to see where each route table belongs. Focus on the default route table and your VPC's route table.
- Let's investigate. Select one of the two route tables and select the **Routes** tab.
- Uncheck that route table, and switch to the other route table.
- Select the **Routes** tab again.
- Aha, the two tables have different routes!
- Let's rename your NextWork VPC route table so it's easier to recognise.
- Make sure you have your NextWork VPC route table selected - this is the route table with a single route to 10.0.0.0/16.
- Select the pencil icon in the **Name** column of your route table.
- Enter the name NextWork route table.
- Select **Save**.

- Select the **Routes** tab.
- Choose **Edit routes.**
- Choose **Add route** near the bottom of the page.
- Destination: 0.0.0.0/0

💡 **Why is the destination 0.0.0.0/0?**
0.0.0.0/0 means **all** IPv4 addresses! When you set 0.0.0.0/0 as the destination in a route table, you are creating a default route that sends any traffic that doesn't match more specific routes on your route table.

In your case, since the the only other route has a destination of 10.0.0.0/16, this means all traffic that is not bound for another resource within your VPC is bound for the internet gateway!

The internet gateway then forwards this traffic to the internet, allowing your resources to communicate with external networks and users.

Extra for Experts: Routing rules are evaluated from the most restrictive (i.e. destinations with the bigger number after the slash) through to the least restrictive (which is 0.0.0.0/0 since it refers to all IPv4 addresses). This means your route table will first try to send traffic within the VPC if the destination falls within the VPC's CIDR block, otherwise it is send to the Internet.

- Target: **Internet Gateway**.
- Select **NextWork IG**.
- Choose **Save changes.**
-
- Choose the **Subnet associations** tab.
- Under the **Explicit subnet associations** tab, choose **Edit subnet associations**
- Select **Public 1**.
- Choose **Save associations.**

Ayyy nice! Your subnet is now public because it is connected to the Internet via the internet gateway!

Create a security group

In this task, let's add a security group so that users can access resources in your VPC.

💡 **What is a security group?**
If VPCs are cities and subnets are neighbourhoods, a security group is a security checkpoint, or security guard, at the entrance for each building (resource) in that neighbourhood (subnet).

Every resource must be associated with a security group. This means security groups don't attach to a VPC or a subnet, they attach to a specific resource within that VPC/subnet. If you don't specify a security group when you launch a resource, it will use the default security group that AWS creates whenever you set up a VPC.

Security groups are responsible for checking who comes in and out. They have strict rules about what kind of traffic can enter or leave the resource based on its IP address, protocols and port numbers.

💡 **Protocols and port numbers? What do they mean?**

- **Protocols: With VPCs as our city and every resource as a building, think of protocols as different vehicles, like buses, taxis and trucks, to deliver data in different ways. Protocols are special rules that help data move across the internet, each designed to send data for a specific kind of task. Here are some protocols you might come across:**
  - **HTTP (Hypertext Transfer Protocol): This is the standard protocol for sending web pages over the internet. Just like buses carry passengers, HTTP carries web page data to your browser. Most website links start with HTTP, for example, http://www.nextwork.org/.**
  - **SMTP (Simple Mail Transfer Protocol): SMTP is the standard protocol used for sending emails across the Internet! It acts as a mailman, ensuring that your email reaches the correct inbox without errors.**
  - **FTP (File Transfer Protocol): FTP is like a delivery truck that is used for transporting large loads of files between computers on a network. It's a popular tool for uploading and downloading bulky items to and from servers, and developers often use FTP to upload website files directly to a hosting server from their local computers.**
  - **SSH (Secure Shell Protocol): SSH is like a secure, private car that allows encrypted communication directly with a server for private tasks such as managing software or configuring settings. In AWS and other cloud environments, SSH is widely used for securely accessing and managing virtual servers like EC2 instances! You'll eventually come across SSH in more advanced projects.**
- **Port numbers: Think of port numbers as specific doors on a building where data will enter or exit. Each door is designed for a specific protocol, helping servers (i.e. EC2 instances) direct incoming and outgoing traffic to the right application/process to process that protocol. Without port numbers, an EC2 instance would struggle to manage incoming data correctly. It wouldn't know which application or service each piece of data should be directed to.**

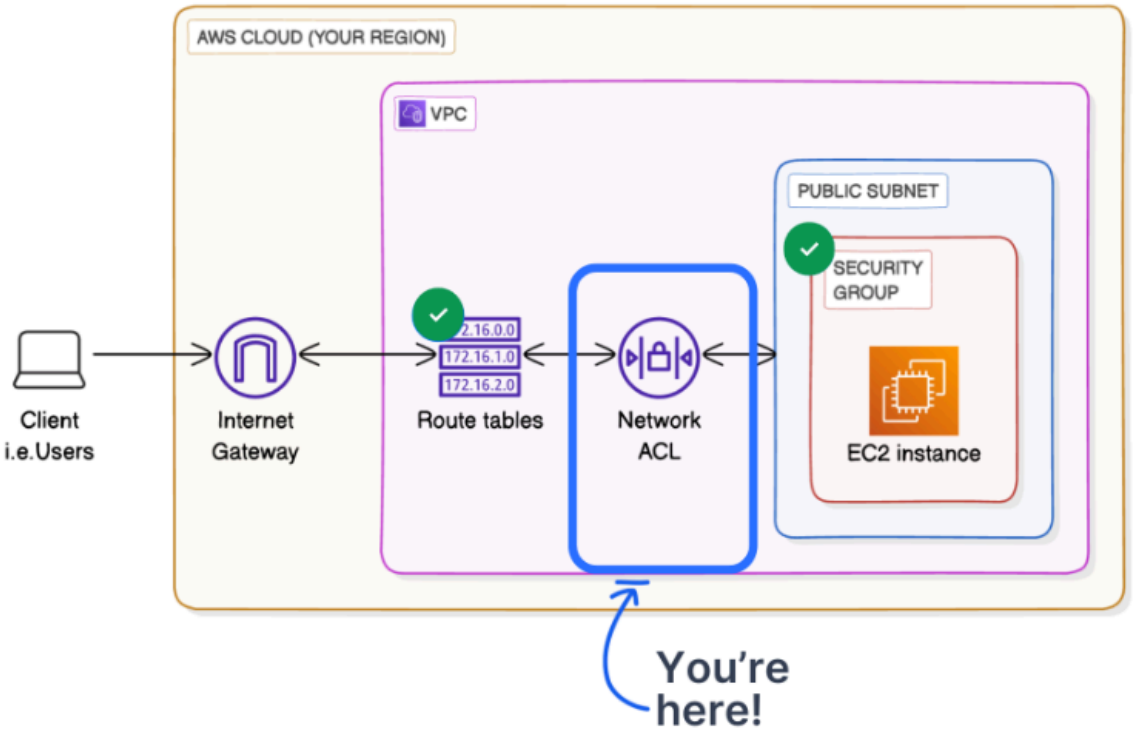
  Some common port numbers are:
  - **Port 80: for HTTP protocols delivering web pages.**
  - **Port 25: for SMTP protocols delivering email.**
  - **Port 21: for FTP protocols delivering files.**
- **In the left navigation pane, choose Security groups. Note that this is further down the navigation pane than our other pages so far!**

💡 Woah! Why do we already have existing security groups?
AWS automatically creates a default security group for each new VPC, which allows all traffic between resources within the same VPC. This default rule enables secure communication between resources without exposing them to external threats!

Extra for Experts: Can you tell which security group belongs to your NextWork VPC? Click into the VPC ID of the security groups to find out!

💡 Is it free to keep this many security groups?
Yup! AWS does not charge for creating and maintaining security groups.

- **Choose Create security group.**
- **Security group name: NextWork Security Group**
- **Description: A Security Group for the NextWork VPC.**
- **VPC: NextWork VPC**
- **Under the Inbound rules panel, choose Add rule.**
  -
- **Type: HTTP**
- **Source: Anywhere-IPv4**
- **At the bottom of the screen, choose Create security group.**



In the left navigation pane, choose Network ACLs.

Choose the network ACL that's associated with your Public 1 subnet, and check out the tabs for Inbound rules and Outbound rules.

To solidify our learnings, let's recreate this set up ourselves in the console! Your default ACL has everything we need, but it's great practice to set up everything from scratch.

- **Select Create new network ACL.**
- **Name: NextWork Network ACL**
- **VPC: NextWork VPC**
- **Select Create network ACL.**
- **Uncheck the default network ACL you've selected.**
- **Select the checkbox next to NextWork Network ACL**
- **Select the Inbound rules tab.**
- **Select Edit inbound rules.**
- **Select Add new rule.**
- **Rule number: 100**

Type: All traffic.

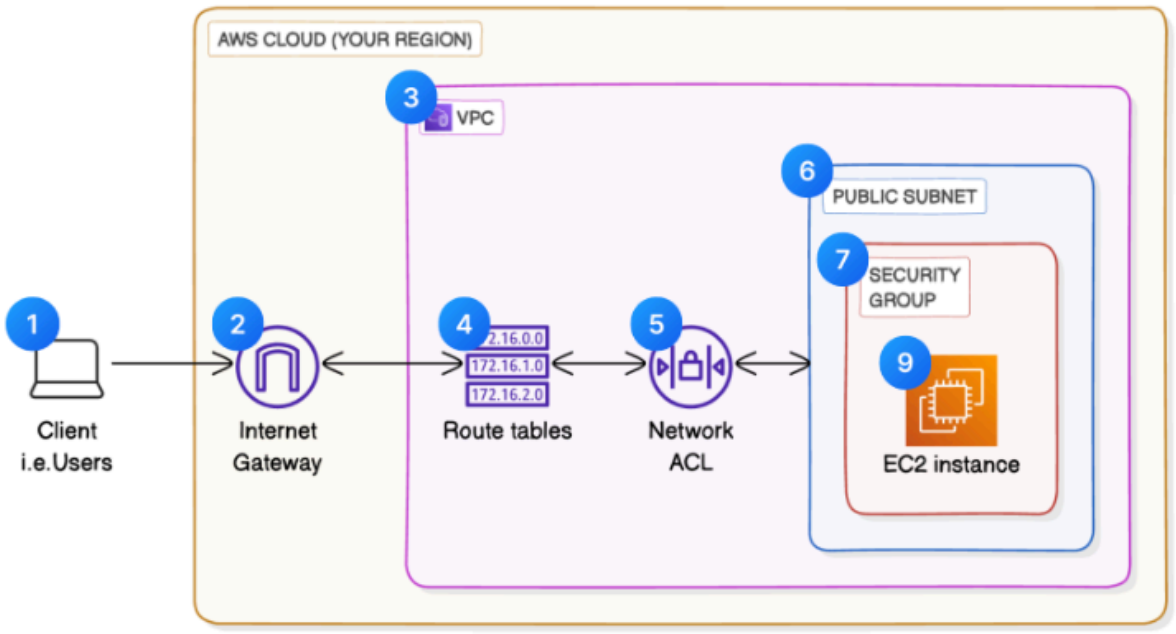
Source: 0.0.0.0/0

Click Save changes.

Select the Subnet associations tab, which should be right next to the Outbound rules tab.

Under the Subnet associations tab, select Edit subnet associations.

- **Select your Public 1 subnet.**
- **Select Save changes.**



1. 💻 **Client/User: A user enters the URL of your website into their web browser and hits enter.**
2. 🚪 **Internet Gateway: The request is sent from the user's browser through the internet and reaches your internet gateway, NextWork IG.**
3. 🌐 **VPC: The internet gateway forwards the user's request to the VPC it's attached to, NextWork VPC.**
4. 🪧 **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 get put on the local route in the route table.**
5. 📋 **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.**
6. 🟥 **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.**

Amazing work - that's heaps of resources that you've created.