# Experiment 1 — Create a VPC with Subnet and Internet Gateway

**Course:** Cloud Computing Lab **Level:** Beginner **Duration:** 30–45 minutes
**Date:** 2025-10-08

## Aim

To create an AWS Virtual Private Cloud (VPC) with a public subnet and configure an Internet Gateway so instances launched in the subnet can access the internet.

## Description

This experiment introduces students to basic AWS networking by creating a VPC, a subnet, an Internet Gateway (IGW), and a route table. A VPC provides an isolated virtual network in AWS. An Internet Gateway enables communication between the VPC and the Internet. By completing this lab, students will understand CIDR addressing, subnetting, and how to allow outbound/inbound internet traffic through route tables and gateways.

## Prerequisites

• An AWS account with permissions to create VPCs, subnets, and Internet Gateways.

• Access to the AWS Management Console (or AWS CLI configured locally).

• Basic familiarity with CIDR notation (example: 10.0.0.0/24).

• A web browser and optional terminal with ssh/aws CLI for verification.

## Procedure — Step by step (Console)

**Step 1:** Log in to the AWS Management Console and open the *VPC* dashboard.

**Step 2:** Create a VPC: Click *Create VPC* → set Name = **MyVPC** → IPv4 CIDR block = **10.0.0.0/24** → Create. After creation, enable DNS hostnames in VPC settings.

**Step 3:** Create a Subnet: In the VPC dashboard choose *Subnets* → *Create subnet*. Select VPC = **MyVPC**. Name = **MySubnet**. CIDR block = **10.0.0.0/25**. Choose an Availability Zone (e.g., us-east-1a). Enable Auto-assign public IPv4 address for the subnet (so launched instances receive public IPs).

**Step 4:** Create an Internet Gateway (IGW): Go to *Internet Gateways* → *Create internet gateway*. Name it **MyIGW**. After creating, select it and click *Attach to VPC* → choose **MyVPC**.

**Step 5:** Create a Route Table: Go to *Route Tables* → *Create route table*. Name = **MyRouteTable**, VPC = **MyVPC**. After creation, select the route table → *Edit routes* → *Add route*: Destination = **0.0.0.0/0** → Target = **MyIGW** → Save.

**Step 6:** Associate the route table with the subnet: In the route table details, choose *Subnet associations* → *Edit subnet associations* → select **MySubnet** → Save.

**Step 7 (Verify):** Confirm that: (a) the VPC shows CIDR 10.0.0.0/24, (b) the subnet MySubnet uses CIDR 10.0.0.0/25 and has Auto-assign public IPv4 enabled, (c) the Internet Gateway MyIGW is attached to MyVPC, and (d) the route table MyRouteTable contains a route 0.0.0.0/0 → MyIGW and is associated with MySubnet.

## Procedure — Equivalent AWS CLI commands (optional)

```
# (Assumes AWS CLI is configured for a region, e.g., us-east-1)
# 1. Create VPC
VPC_ID=$(aws ec2 create-vpc --cidr-block 10.0.0.0/24 --query 'Vpc.VpcId' --output text)
aws ec2 create-tags --resources $VPC_ID --tags Key=Name,Value=MyVPC
aws ec2 modify-vpc-attribute --vpc-id $VPC_ID --enable-dns-hostnames

# 2. Create subnet
SUBNET_ID=$(aws ec2 create-subnet --vpc-id $VPC_ID --cidr-block 10.0.0.0/25 --availability-zone us-
aws ec2 create-tags --resources $SUBNET_ID --tags Key=Name,Value=MySubnet
aws ec2 modify-subnet-attribute --subnet-id $SUBNET_ID --map-public-ip-on-launch

# 3. Create and attach Internet Gateway
IGW_ID=$(aws ec2 create-internet-gateway --query 'InternetGateway.InternetGatewayId' --output text)
aws ec2 create-tags --resources $IGW_ID --tags Key=Name,Value=MyIGW
aws ec2 attach-internet-gateway --internet-gateway-id $IGW_ID --vpc-id $VPC_ID

# 4. Create route table and route to IGW
RTB_ID=$(aws ec2 create-route-table --vpc-id $VPC_ID --query 'RouteTable.RouteTableId' --output tex
aws ec2 create-tags --resources $RTB_ID --tags Key=Name,Value=MyRouteTable
aws ec2 create-route --route-table-id $RTB_ID --destination-cidr-block 0.0.0.0/0 --gateway-id $IGW_
aws ec2 associate-route-table --route-table-id $RTB_ID --subnet-id $SUBNET_ID
```

## Expected Result

After completing the steps, students should have a VPC named **MyVPC** (10.0.0.0/24), a subnet **MySubnet** (10.0.0.0/25) with Auto-assign Public IPv4 enabled, an Internet Gateway **MyIGW** attached to the VPC, and a route table **MyRouteTable** that routes 0.0.0.0/0 to the IGW. Instances launched into MySubnet with a public IP will be able to reach the Internet and be reached from the Internet (subject to security group rules).

## Verification / Checklist

• VPC 'MyVPC' exists with CIDR 10.0.0.0/24.

• Subnet 'MySubnet' exists in MyVPC with CIDR 10.0.0.0/25 and 'Auto-assign public IPv4 address' enabled.

• Internet Gateway 'MyIGW' is attached to 'MyVPC'.

• Route table 'MyRouteTable' has a route 0.0.0.0/0 pointing to 'MyIGW' and is associated with 'MySubnet'.

## Tips & Troubleshooting

• If launched instances do not have internet access, confirm the subnet has a public IP and the route table contains 0.0.0.0/0 → IGW.

• Restrict SSH access in security groups to your lab IP range instead of 'Anywhere' for safety.

• If using the AWS CLI, ensure you are operating in the intended region (check AWS_DEFAULT_REGION).

Instructor: Cloud Computing Lab • Prepared by: ChatGPT

Generated on 2025-10-08 15:24:26