# Creating an Infrastructure in AWS

T.Sri Poojith[1] and Mr. P.Muthyalu[2]

[1](Department of CSE, Narayana Engineering Collenge, Gudur)

[2](Assistant Professor, Department of CSE, Narayana Engineering Collenge, Gudur)

***Abstract:*** *This paper outlines the process of building a secure and scalable infrastructure on Amazon Web Services (AWS) using core services such as Virtual Private Cloud (VPC), subnets, route tables, security groups, internet gateways, and EC2 instances. AWS provides a comprehensive cloud platform that allows users to create a robust and flexible infrastructure tailored to their specific needs. We will detail the configuration steps for each component, starting with the creation of a VPC to establish a virtual network, followed by the addition of subnets to segment the network. Route tables will be configured to manage traffic flow, and security groups will be established to enforce firewall rules and protect resources. An internet gateway will be connected to enable external connectivity, and finally, EC2 instances will be deployed to provide computational power. Throughout the process, we will emphasize security best practices to ensure that the infrastructure is not only functional but also secure against potential threats. This guide is intended for beginners or those new to deploying infrastructure on AWS, providing a clear and practical approach to setting up a foundational cloud environment that can support a variety of use cases, from web hosting to development and testing environments.*

***Keywords:*** *Amazon Web Services (AWS),Virtual Private Cloud (VPC),Public Subnet ,Route Table, Security Group, Internet Gateway,EC2 Instance, Cloud Computing*.

## I. INTRODUCTION

Amazon Web Services (AWS) provides a comprehensive cloud platform that facilitates the deployment of a wide range of services and infrastructure configurations. This paper outlines the steps involved in setting up a basic yet effective infrastructure utilizing core AWS services. These services include a Virtual Private Cloud (VPC), a public subnet, a route table, a security group, an Internet gateway, and an EC2 instance.

The process begins with the creation of a VPC, which serves as a virtual network dedicated to your AWS resources. Following this, a public subnet is configured within the VPC to allow for internet-accessible resources. The next step involves setting up a route table to control the routing of network traffic within the subnet. To secure the infrastructure, a security group is established, defining specific firewall rules for the EC2 instance, which acts as the virtual server. An Internet gateway is then attached to the VPC, enabling internet connectivity for the resources within the public subnet. Finally, an EC2 instance is launched within this subnet, providing a flexible environment suitable for various purposes, including web hosting, development, and testing activities.

By detailing these steps, this paper aims to provide a clear and practical guide for beginners or those new to AWS, helping them to build a foundational cloud environment that is both secure and scalable.

## II. COMPONENTS AND METHODOLOGY:

**1.Virtual Private Cloud(VPC):** A Virtual Private Cloud (VPC) is an isolated network environment within AWS that allows you to launch and manage resources in a logically separated virtual network. It provides control over your networking setup, including IP address range selection, subnet creation, and configuration of route tables and gateways. This isolation ensures secure and private operation of AWS resources, with advanced security features like network ACLs and security groups to manage traffic.

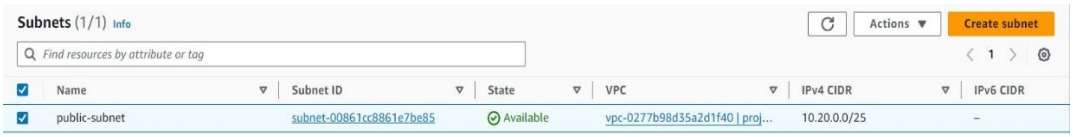
Configuration Steps:

1. Create a VPC: Specify an IPv4 CIDR block such as 10.20.0.0/22.
2. Tagging: Name your VPC for easy identification.


**2.Public Subnet:** A public subnet is a subdivision within your VPC that consists of a range of IP addresses and is designed to be accessible from the internet. Resources deployed within a public subnet can directly communicate with the internet through an attached Internet Gateway. This configuration is typically used for resources that need to be publicly accessible, such as web servers or application servers, ensuring they can send and receive data to and from the internet. The public subnet setup includes assigning a public IP address to each resource, allowing it to be reachable by external users and services.

Configuration Steps:

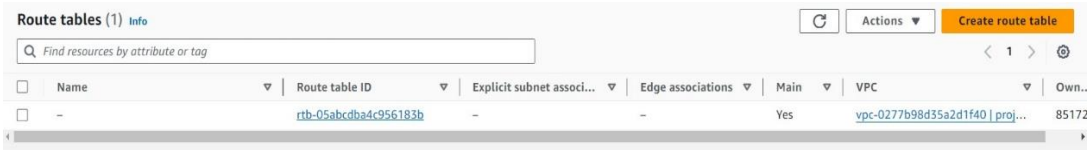
1. Create Subnet: Choose the VPC and specify a subnet CIDR block (e.g., 10.0.1.0/24).
2. Enable Auto-assign Public IP: Ensure instances launched in this subnet receive a public IP address.




**3.Route Table:** A route table is a critical component within your VPC that contains a set of rules, known as routes, which dictate how network traffic is directed. These rules specify the paths that data packets should take to reach various destinations, both within your VPC and outside it. By configuring a route table, you can control the flow of traffic to ensure it reaches the intended subnet, network interface, or gateway. This allows for efficient and organized routing of traffic, supporting both internal communication between AWS resources and external communication with the internet or other networks.

Configuration Steps:

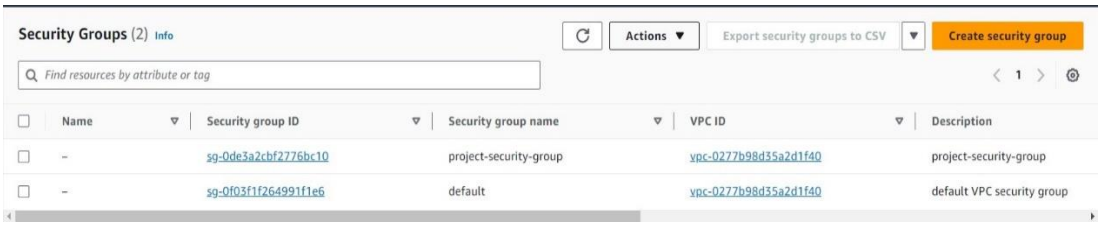
1. Create Route Table: Associate it with your VPC.
2. Add Routes: Add a route that directs internet traffic (0.0.0.0/0) to the Internet gateway.




**4.Security Group:** A security groups act as virtual firewalls, controlling the inbound and outbound traffic flow to and from your resources. They function at the instance level, filtering traffic at the port and protocol level.

Configuration Steps:

1. Create Security Group: Associate it with your VPC.
2. Define Rules: Allow inbound SSH (port 22) and HTTP (port 80) traffic, and all outbound traffic.




**5.Internet Gateway:** An Internet Gateway is a vital component that enables communication between instances within a VPC and the internet. It acts as a bridge, facilitating the two-way flow of traffic from the internet to the VPC and vice versa. By attaching an Internet Gateway to your VPC, resources such as EC2 instances within public subnets can access external services, send data, and receive responses from the internet. This setup is essential for deploying applications and services that require internet connectivity, ensuring seamless integration between your AWS resources and the global network.

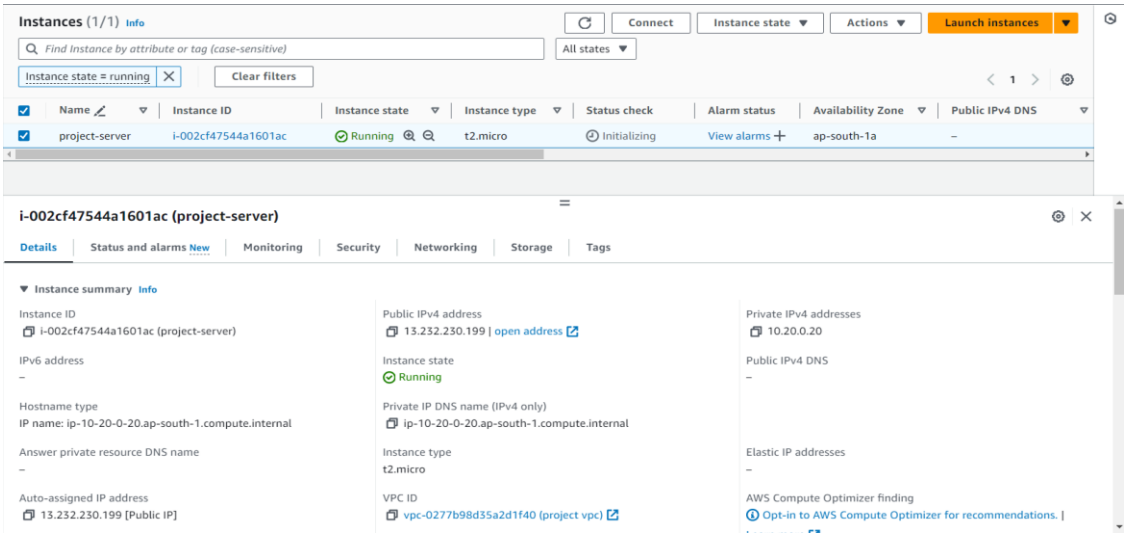
Configuration Steps:

1. Create Internet Gateway: Attach it to your VPC.
2. Update Route Table: Modify the route table to point internet-bound traffic to the Internet gateway.


**6.EC2 Instance:** An EC2 (Elastic Compute Cloud) instance is a fundamental building block of AWS that offers scalable computing capacity in the cloud. It allows users to launch virtual servers, known as instances, in a variety of configurations based on their computing needs. EC2 instances provide on-demand access to computational resources, including processing power, memory, and storage, allowing users to deploy and manage applications and workloads with flexibility and efficiency. With EC2, users can easily scale their computing resources up or down as needed, paying only for the resources they consume, making it an ideal choice for a wide range of use cases, from hosting websites to running complex applications.

Configuration Steps:

1. Launch Instance: Select a Amazon Machine Image (AMI), instance type, and configure instance details including the public subnet and security group.
2. Key Pair: Create or select a key pair for SSH access.




**7.Web Server Setup:** Configure the EC2 instance to run a small web server.

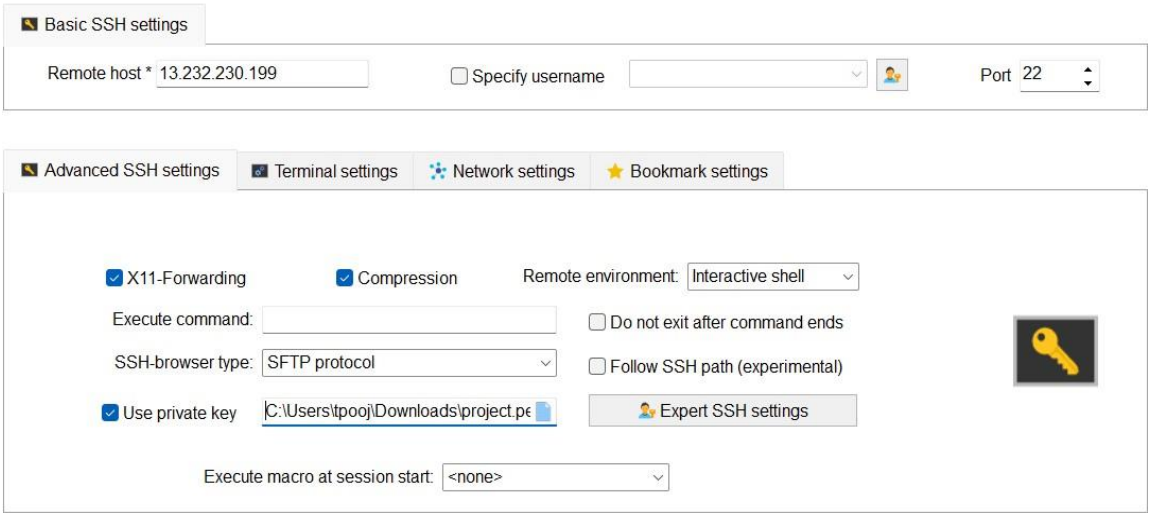
Configuration Steps:

1. Connect to Instance: Use SSH to connect the instances.
2. Install Web Server: Install and configure a web server (e.g., Apache or Nginx).

## III. INSTALLATION WEBSERVER ON EC2 INSTANCE:

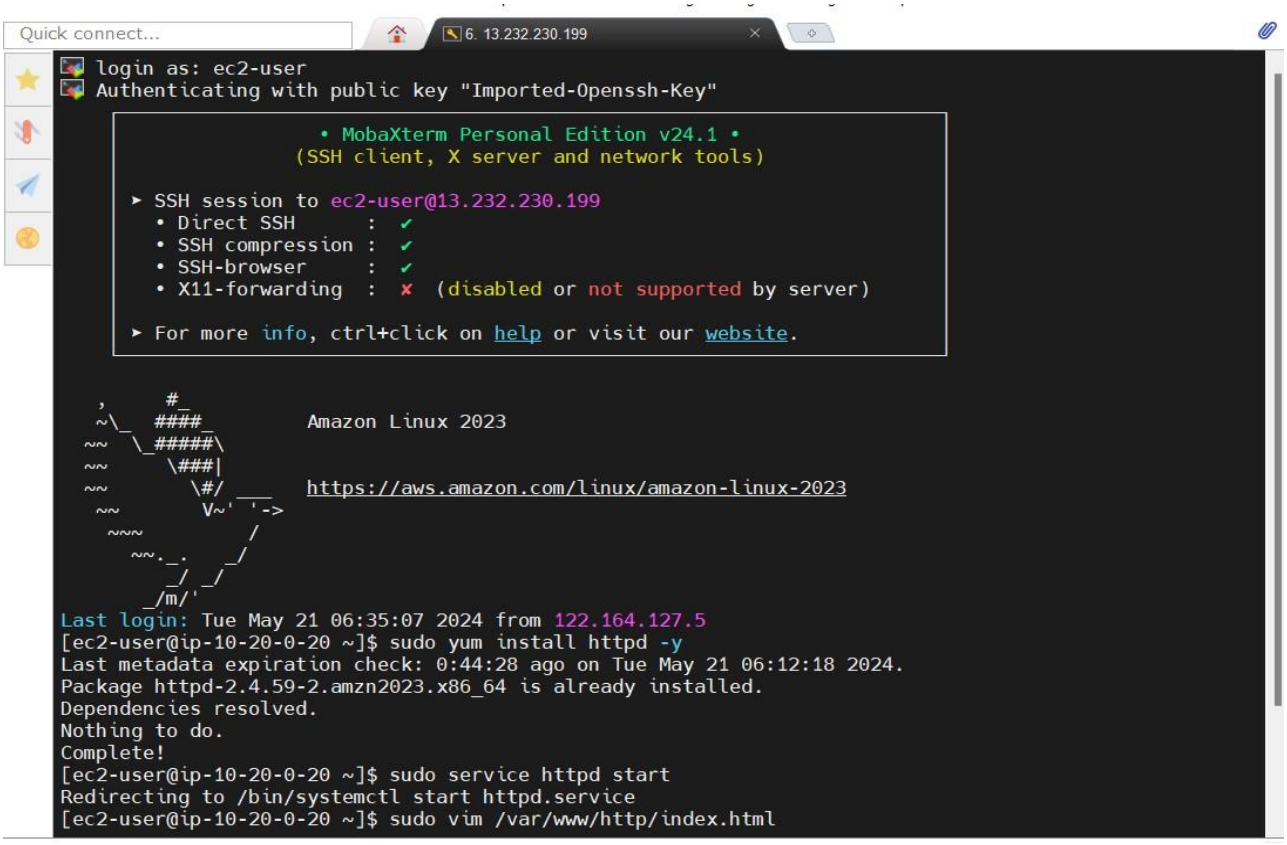
### 1.Connect to EC2 Instance:

1. Open MobaXterm.
2. Click on "Session" in the top left corner.
3. Select "SSH."
4. Enter the Public IP address of your EC2 instance.
5. In the "Specify username" field, enter ec2-user.
6. Under "Advanced SSH settings," select your private key file (MyKeyPair.pem).
7. Click "OK" to connect.



## 2.Install Apache WebServer:

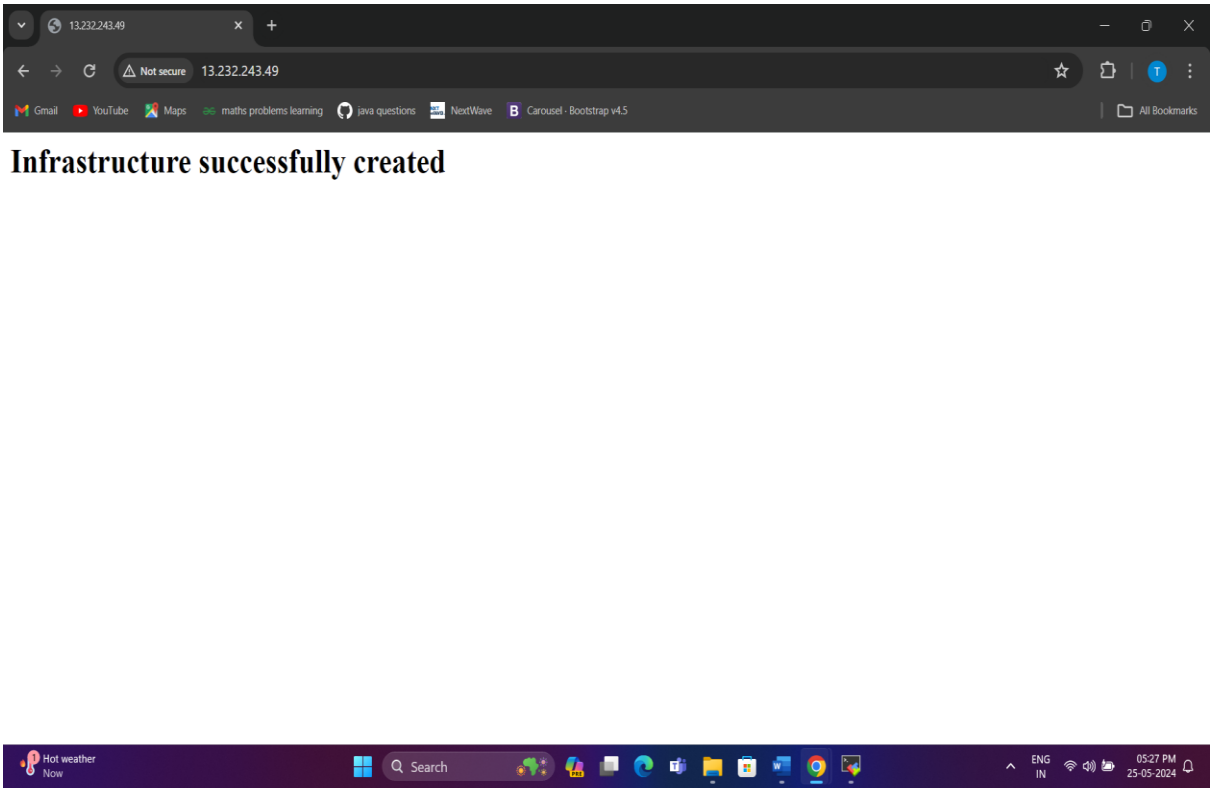
1. Once connected to EC2 instance, update package manager connect to your EC2 instance and update the package manager using the command:
2. sudo yum update -y
3. Once the package manager is updated, install Apache using the command: sudo yum install httpd -y
4. After installation, start the Apache service with the following command:
   sudo systemctl start httpd
5. To ensure Apache starts automatically on boot, run the command:
6. sudo systemctl enable httpd
7. Verify Installation:
8. you can verify that Apache is running correctly by accessing your EC2 instance's public IP address or DNS name in a web browser. You should see the default Apache welcome page.
9. Customize Default Page (Optional): If you want to customize the default Apache landing page, you can replace the content of the **/var/www/html/index.html** file with your desired HTML content.
10. For example:
11. echo "<h1>Infrastructure Successfully Created</h1>" | sudo tee /var/www/html/index.html



## IV. VERIFY WEB SERVER

1.Open a web browser and enter the public IP address of your EC2 instance.

2.You should see the message “Infrastructure Successfully Created” displayed on the web Page.




## V. CONCLUSION

Building an infrastructure in AWS involves integrating various components to create a secure, scalable, and efficient environment. This paper has detailed the essential steps to set up a Virtual Private Cloud (VPC), public subnet, route table, security group, Internet gateway, and an EC2 instance running a web server. Each component is crucial in establishing a foundational cloud infrastructure suitable for multiple applications, including web hosting, development, and testing. By following these guidelines, users can develop a robust AWS environment that balances functionality and security. Future enhancements could include the addition of private subnets for internal resource segregation, advanced security measures, and complex network configurations to optimize performance. Expanding the infrastructure with features like automated scaling, load balancing, and advanced monitoring will further enhance its resilience and efficiency, ensuring it meets evolving demands and challenges.

## VI. REFERENCES

1. Amazon Web Services: https://aws.amazon.com/what-is-aws/
2. cloud service models: https://www.appviewx.com/education-center/cloud-services/
3. Virtual private network(VPC): https://docs.aws.amazon.com/vpc/
4. Subnet: https://docs.aws.amazon.com/vpc/latest/userguide/configure-subnets.html
5. Route Table: https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Route_Tables.html
6. Security group: https://docs.aws.amazon.com/vpc/latest/userguide/vpc-security groups.html
7. EC2 instance: https://docs.aws.amazon.com/ec2/