## Deployment of a React App on an EC2 Instance

This documentation provides a step-by-step guide to deploying a React application on an Amazon EC2 instance. Follow these instructions to set up your EC2 instance, configure the environment, build your React app, and serve it using Nginx.

### Prerequisites

## AWS account

## Basic knowledge of AWS EC2

## React application ready for deployment

## SSH client (e.g., Terminal on macOS/Linux, PuTTY on Windows, MobaXterm )

### Step 1: Set Up an EC2 Instance

1. **Launch an EC2 Instance:**

* **Log in** to your AWS Management Console.
* Navigate to the **EC2 dashboard** and click on "Launch Instance."
* Choose an **Ubuntu Machine Image (AMI)**. For simplicity, use the Amazon Linux 2 AMI.
* And entre the your instance name ex.. React-app
* Choose an **instance type** (e.g., t2.micro for free tier eligibility).
* Configure the instance details, add storage, and add tags as needed.
* Configure the **security group** to allow HTTP (port 80) and SSH (port 22) access.
* **Launch** the instance and download the key pair (.pem file) for SSH access.

2.Connect to Your Instance:

Open your terminal (or PuTTY and MobaXtrem on Windows) and connect to your instance using the SSH command: ssh -i /path/to/your-key-pair.pem ec2-user@your-instance-public-dns

### Step 2: Configure the Environment

**1.Update the Package Lists:** sudo apt update -y

**2. Install required package**: sudo apt-get install -y ca-certificates curl gnupg

**3. Create the Keyrings Directory:** sudo mkdir -p /etc/apt/keyrings

4. Download the GPG Key and Save It: curl -fsSL https://example.com/key.gpg | sudo tee /etc/apt/keyrings/example.gpg > /dev/null

### Step 3: Update Package Lists:

sudo apt-get update –y

## install node.js

sudo apt-get install -y nodejs

node –v for the check the installed version of Node.js on your system. This is a quick and easy way to verify that Node.js is installed correctly.

npm –v (Node Package Manager) After installing Node.js and npm, running confirms that npm is installed correctly.

Sudo apt install npm

Npm –v

### Step 4: Clone and Build Your React App:

git clone <https://github.com/africacloudpapa/new-react-app.git>

The git clone command creates a local copy of a remote repository. This is useful for starting development on an existing project, contributing to an open-source project.

ls List Files and Directories

Cd new-react-app/ used to change the current working directory to a directory name.

ls

npm install

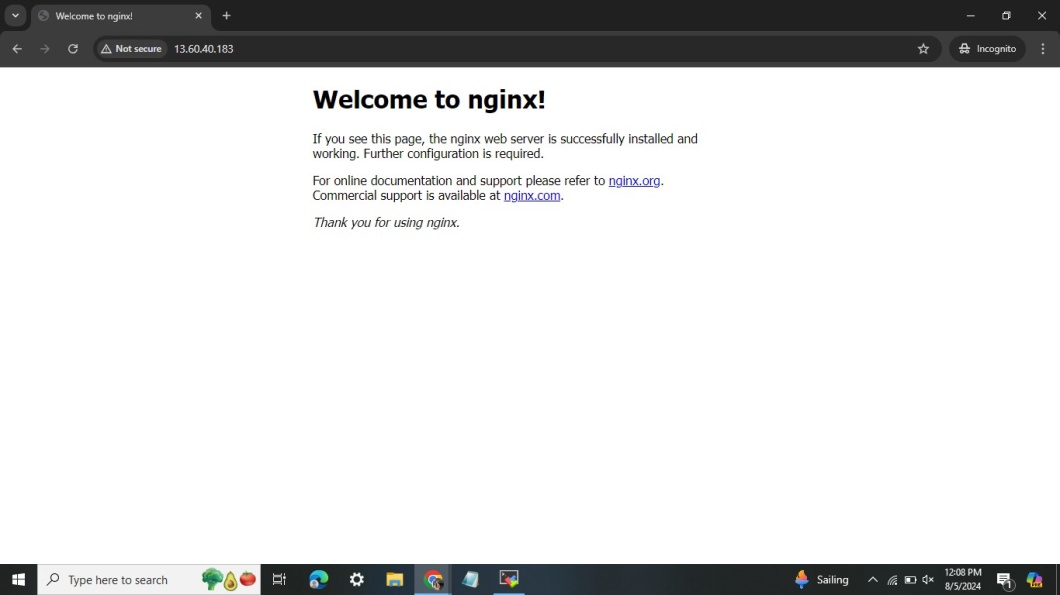
npm run build

sudo apt install nginix –y

ls

ls -l build/

ls -l /var/www/html/index.nginx



after above command you can see this output….

Sudo rm /var/www/html/index.nginx debian html

ls -l /var/www/html/

sudo cp -r build/\* /var/www/html/ they are used in the context of copying files and verifying the results.

ls –l /var/www/html/

check the result using your instance public Ip

