Deploy Mern Stack Application

Prepared by:

Tehzeeb ul Hassan

Roll No:

IAC-FA22-BCS-009

Project Overview:

This project demonstrates the deployment of a MERN (MongoDB, Express, React, Node.js) stack application using Infrastructure as Code (IaC) principles.

Submission Date:

October 11, 2024

Course:

DevOps

Institution:

Institute for Art and Culture

Repository Link:

https://github.com/tehzeeb45/iac-final-project-mern-stack.git

DOUMENTATION

Linux

1. What is Linux?

Linux is an open-source operating system kernel that forms the core of many distributions (distros) like Ubuntu, CentOS, Fedora, and others. It manages hardware resources and provides services for software applications.

2. Linux Directory Structure

/: Root directory, the base of the filesystem.

/home: User home directories (e.g., /home/username).

/etc: Configuration files for system and applications.

/bin and /sbin: Essential user and system binaries.

/var: Log files, spools, and data that changes frequently.

/usr: User binaries and applications.

/tmp: Temporary files.

3. How to Install Packages (git, vim ,Jenkins, nginx, jdk, node latest)

- **Update Package Lists:**
  sudo apt update
- **Install git:**
  sudo apt install git
- **Install vim:**
  sudo apt install vim
- **Install Jenkins:**
  sudo apt install Jenkins
- **Install Nginx:**
  sudo apt install nginx

- **Install JDK:**
  sudo apt install openjdk-21-jdk
- **Install the Latest Node.js:**

  curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.0/install.sh | bash

  export NVM_DIR="$HOME/.nvm"

  [-s "$NVM_DIR/nvm.sh"] && \. "$NVM_DIR/nvm.sh" # This loads nvm

  [-s "$NVM_DIR/bash_completion"] && \. "$NVM_DIR/bash_completion"

  nvm install 20
- **verifies the right Node.js version is in the environment**

  node -v # should print `v20.18.0`
- **verifies the right npm version is in the environment**

  npm -v # should print `10.8.2`

4. Uninstall Each Package

- **Remove git:**

  sudo apt remove --purge git
- **Remove vim:**

  sudo apt remove --purge vim
- **Remove Jenkins:**

  sudo apt remove --purge Jenkins
- **Remove Nginx:**

  sudo apt remove --purge nginx
- **Remove JDK :**

  sudo apt remove --purge openjdk-21-jdk
- **Remove Node.js:**

  sudo apt remove --purge nodejs

5. How to create users?

- sudo adduser hassan

6. How to Set a Password

- sudo passwd hassan

7. Check All Users

- cut -d: -f1 /etc/passwd

8. How to Expire a Password

- sudo passwd --expire hassan

9. How to Delete a User and Home Directory

- sudo deluser --remove-home Hassan

10. How to Add a User to the Sudo Group

- sudo usermod -aG sudo hassan

11. How to SSH

- ssh tehzeeb@192.168.1.112

12. How to Change SSH Port from 22 to 8888

- **Edit the SSH configuration file:**

  sudo nano /etc/ssh/sshd_config

- **Find the line that says #Port 22, and change it to:**

  Port 8888

- **Restart the SSH service:**

  sudo systemctl restart ssh

13. How to Give Permissions to Users?

- **Give read, write, and execute permissions**

  sudo chmod u+rwx filename

14. How to Check Open Ports?

- sudo netstat -tuln

Git

1. **What is VCS (Version Control System)**

   A Version Control System (VCS) is a tool that helps manage changes to files, typically source code. It allows multiple users to collaborate on a project by tracking changes, enabling version tracking, and facilitating the recovery of earlier versions of files.

2. **What is Git?**

Git is the most commonly used version control system. Git tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to.

1. **Why We Use Git?**

- **Collaboration**: Multiple developers can work on the same project without conflicts.
- **Version History:** Every change is recorded, allowing users to track and revert to previous versions.

2. **How to Clone a Repository**

To clone a repository means to create a local copy of a remote repository on your machine. Use the following command:

```
git clone <repository-url>
```

3. **How to Pull Changes**

```
git pull
```

4. **What is Git Commit and Checkout?**

- **git commit:** Records changes to the local repository.

```
git commit -m "Your commit message"
```

- **git checkout:** To switch to a branch:

```
git checkout branch-name
```

5. **How to Push Changes**

To send your committed changes to a remote repository, use

```
git push
```

6. **How to Switch from One Branch to Another**

```
git checkout branch-name
```

7. **Difference Between Git and GitHub, GitLab, Bitbucket?**

- **Git**: A version control system for tracking changes in files. It is a command-line tool that operates on your local machine.
- **GitHub:** A web-based platform that hosts Git repositories. It provides a user-friendly interface, collaboration tools, and additional features like issue tracking and project management.
- **GitLab:** Similar to GitHub, GitLab is a web-based platform that hosts Git repositories but also includes integrated CI/CD pipelines and project management features. It can be self-hosted.
- **Bitbucket:** Another web-based platform for hosting Git (and Mercurial) repositories. It offers features like pull requests and integrations with other tools, including Atlassian products like Jira

8. **Steps to Change the SSH Configuration and Update Password in a VM**

1. Navigate to the SSH Configuration Directory:

- First, access the SSH configuration files.
- cd /etc/ssh/
- ls

2. Navigate to sshd_config.d Directory:

- cd sshd_config.d
- ls

3. Edit SSH Configuration Files:

- sudo vim 60-some-config.conf

4. You can add or modify SSH configuration options like:

- PasswordAuthentication yes

5. Change the Password for the VM User:

- sudo passwd ubuntu

6. Restart the SSH Service:

- sudo systemctl restart ssh

CI/CD

1. **What is CI / CD**

Continuous Integration (CI) and Continuous Deployment (CD) are software development practices that aim to automate and improve the process of software delivery.

- **Continuous Integration (CI):** The practice of frequently integrating code changes into a shared repository, allowing for automated testing and validation of the changes. This helps detect issues early in the development cycle.
- **Continuous Deployment (CD):** The process of automatically deploying code changes to production after they pass automated tests. This allows for more frequent releases and faster delivery of features to users.

2. **Tools Used for CI / CD**

- **GitLab CI/CD**
- **Jenkins**
- **Codeship**
- **TeamCity**
- **Buddy**

3. **What is the root path of Jenkins, where all pipeline and configuration and available**

/var/lib/Jenkins

4. **Command Check Jenkins is in the Docker group:**
    - sudo usermod -aG docker jenkins

- sudo systemctl restart jenkins

Clouds

1. **What are Clouds**

   Clouds refer to a network of remote servers hosted on the Internet that store, manage, and process data, rather than a local server or a personal computer. Cloud computing offers scalable resources and services on-demand.

   Four Major Cloud Providers

   - **Amazon Web Services**
   - **Microsoft Azure**
   - **Google Cloud**
   - **Alibaba Cloud**
   - **IBM**
   - **Workday**

2. **What is AWS?**

   Amazon Web Services (AWS) is a comprehensive cloud computing platform provided by Amazon, offering a wide range of services such as computing power, storage options, and networking capabilities. AWS allows businesses to host applications and services in the cloud, providing scalability and flexibility**.**

3. **What is EC2**
   (EC2) is a web service that provides resizable compute capacity in the cloud. It allows users to launch and manage virtual servers (instances) on-demand, offering flexibility and control over computing resources**.**

4. **What is Security Group**

   A Security Group acts as a virtual firewall for EC2 instances, controlling inbound and outbound traffic. It allows users to define rules to permit or restrict access to their instances based on IP address, protocol, and port number

5. **What is keypair**

   A Key Pair consists of a public key and a private key used to securely access EC2 instances. The public key is stored on the instance, while the private key is kept by the user. SSH access is typically established using this key pair

6. **What is AMI**

   An Amazon Machine Image (AMI) is a pre-configured template used to create EC2 instances. It contains the operating system, application server, and applications necessary for launching an instance. Users can create their own AMIs or choose from AWS's catalog**.**

7. **What is S3 bucket?**

An Amazon S3 (Simple Storage Service) Bucket is a scalable storage resource used to store and retrieve any amount of data at any time.

8. **What is IAM?**

IAM (Identity and Access Management) is a service offered by AWS to securely control access to AWS services and resources. It allows administrators to manage permissions and define who can access specific resources and what actions they can perform. IAM helps ensure that the right people have the right access to the right resources in a scalable and secure way**.**

9. **What are Roles in IAM?**

AWS Identity and Access Management (IAM) roles are entities you create and assign specific permissions to that allow trusted identities such as workforce identities and applications to perform actions in AWS**.**

10 . **What are Policies in IAM?**

IAM is a framework of policies and technologies to ensure that the right users have the appropriate access to technology resources. An AWS IAM policy defines the permissions of an identity (users, groups, and roles) or resource within the AWS account.

Configuration Management Tools

1. **What are Configuration Management Tools?**

Configuration Management Tools are systems or software used to automate the deployment, management, and maintenance of the desired state of IT infrastructure.

Popular Configuration Management Tools:

- **Ansible**
- **Puppet**
- **Chef**
- **SaltStack**

2. **What is Ansible?**

Ansible is an open-source automation tool used for configuration management, application deployment, orchestration, and provisioning. It uses simple, human-readable YAML syntax to define tasks and configurations.

3. **On which Protocol Does Ansible Work?**

Ansible uses SSH protocol to connect to servers and run tasks**.**

4. **What is inventory file?**

The Inventory File is a file that contains the list of hosts or groups of hosts that Ansible will manage. It tells Ansible where the target machines are and how to connect to them.

Example of Inventory File:

[iac_demo]

192.168.12.131 ansible_user=tehzeeb ansible_ssh_pass=hassan1214

[iac_demo1]

13.60.24.167 ansible_user=ubuntu ansible_ssh_pass=hassan1214

[iac_demo2]

13.60.18.197 ansible_user=ubuntu ansible_ssh_pass=hassan1214

Terraform

1. **What is Infrastructure as Code (IaC)?**

   Infrastructure as code (IaC) is the ability to provision and support your computing infrastructure using code instead of manual processes and settings. Any application environment requires many infrastructure components like operating systems, database connections, and storage.

2. **What Are Different Tools for IaC?**

   Several tools are available for implementing Infrastructure as Code. Some of the most popular tools include

   - **Terraform**:
   - **AWS CloudFormation**
   - **Ansible**:
   - **Puppet**:
   - **Chef**:

3. **What is Terraform?**

   Terraform allows you to describe your complete infrastructure in the form of code. Even if your servers come from different providers such as AWS or Azure, Terraform helps you build and manage these resources in parallel across providers.

4. **What Are Resources in Terraform?**

   Resources in Terraform represent the actual components of your infrastructure, such as virtual machines, storage, networks, security groups, etc

   Example of a Resource Block:

   ```
   resource "aws_instance" "my_ec2" {
   ami           = "ami-12345678"
   instance_type = "t2.micro"
   }
   ```

5. **What Are Providers in Terraform?**

Providers in Terraform are plugins that allow Terraform to interact with APIs of cloud providers or other services (e.g., AWS, Azure, GCP). Providers manage the lifecycle of the resources (create, read, update, delete).

Examples of Popular Providers:

- AWS Provider
- Azure Provider
- Google Cloud Provider:

Networking

1. **What is IP Addressing?**

   IP Addressing is a system used to assign unique numerical labels to devices (e.g., computers, servers, routers) connected to a network. An IP address identifies a device's location on a network and enables communication between devices.

2. **What is a Public and Private IP?**

- **Public IP:** An IP address that is accessible over the internet. Devices with public IPs can be directly accessed from anywhere in the world (e.g., websites, servers). ISPs assign public IP addresses.
- **Private IP**: An IP address used within a private network, such as your home or office network. Devices with private IPs can only communicate with each other within the local network.

3. **What are Ports?**

   A port is a virtual point where network connections start and end. Ports are software-based and managed by a computer's operating system. Each port is associated with a specific process or service. Common ports include:

   - **Port 80**: HTTP (web traffic).
   - **Port 443**: HTTPS (secure web traffic).
   - **Port 22**: SSH (secure shell).

4. **What is NAT and PAT?**
   - **NAT (Network Address Translation):**
   - NAT is a networking technique used to translate private IP addresses within a local network into a public IP address for communication over the internet. It allows multiple devices on a local network to share a single public IP address, helping conserve the number of public IPs needed and enhancing security by hiding internal IP addresses from external networks**.**
   - **PAT (Port Address Translation)**
   - Definition: PAT is a type of NAT that enables multiple devices on a local network to share a single public IP address by using different port numbers. This allows the router to differentiate between multiple connections from different devices, ensuring that incoming and outgoing traffic is correctly routed to the appropriate device within the local network**.**
   -

5. **How to Test Connectivity (Ping, Telnet)?**
    - Command: ping <IP address> (e.g., ping 8.8.8.8)
    - Command: telnet <IP address> <port> (e.g., telnet 192.168.1.1 80).
6. **What is Port Forwarding?**

    Port forwarding is a networking technique used to allow external devices to access services on a private network. It works by redirecting communication requests from one IP address and port number combination to another. This is commonly used for applications like online gaming, web servers, or remote desktop connections.

7. **How to Check IP Address?**

    Ip add

8. **What is TCP and UDP?**

    TCP: Whenever will do data transmission everytime before data transmission it will establish a connection. Data transmission will start after connection has been established

    - **Reliable:** Ensures that data packets are delivered in the correct order and without errors. If a packet is lost, it is retransmitted**.**

    UDP: No connection is established before sending data.

    - **Unreliable:** There is no guarantee that packets will be delivered, nor that they will arrive in order. Lost packets are not retransmitted**.**

DOCKER

1. **What is Docker?**

    Docker is an open-source platform that automates the deployment, scaling, and management of applications using containerization. Containers are lightweight, portable, and ensure that applications run consistently across different environments.

2. **What is an Image**?
    - It is a kind of ready to use software read only template
    - Images is made with souce codes libraries external dependencies and tools.
    - Images can not be update
    - If you want to make change in image you have to create new image
    - Images can not run directly.

Image example

- Application tools and Libraries.
- Node.js setup
- React setup
- Code of application
- Dependencies

- Any other supporting tools
- We can say that images is a template of project.

3. **What is container?**

   The container is a process that runs application with images

4. **What is a Volume?**

   A volume is a persistent storage mechanism used by Docker to store data generated by and used by Docker containers. Volumes allow data to persist even when the container is stopped or removed**.**

5. **What is a Network?**

   Docker networks enable containers to communicate with each other and with external systems. Docker provides different network drivers, such as bridge, host, and overlay, to manage container networking.

6. **What is Dockerfile?**

   A Dockerfile is a text file that contains a series of instructions and commands to automate the process of building a Docker image. It defines the environment and configuration needed for the application.

7. **Differences between COPY and ADD:**
   - **COPY**
   - Used to copy files/directories from the host filesystem to the container filesystem.
   - Does not support remote URLs or automatic extraction of compressed files.
   - **ADD**
   - Similar to COPY, but also supports extracting compressed files (e.g., .tar, .gz) and can copy files from remote URLs.
   - It's generally recommended to use COPY unless you need the specific functionality provided by ADD**.**
8. **Dockerfile Keys Explained?**
   - **FROM:**
   - Specifies the base image to use for the new image. It is the first instruction in a Dockerfile and defines the starting point of the image.
   - **RUN**:
   - Often used for installing packages or software
   - **WORKDIR**
   - Sets the working directory
9. **Start an Ubuntu Container?**

   sudo docker run -it ubuntu /bin/bash

10. **Install JDK, Jenkins, Git, and Nginx?**
    - **Update package list**

      apt-get update
    - **Install required packages**

      apt-get install -y openjdk-11-jdk git

- **Install Jenkins**

  wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | apt-key add -
  echo deb http://pkg.jenkins.io/debian-stable binary/ >
  /etc/apt/sources.list.d/jenkins.list
  apt-get update
  apt-get install -y jenkins
- **Install Nginx**

apt-get install -y nginx
- **Clean up**

apt-get clean && rm -rf /var/lib/apt/lists/*

11. **Create image from running container**
    - Sudo docker commit ubuntu my-ubuntu-image
12. **Start container from above created image and assign port 8080:8080 and access it from browser**
    - sudo docker run -it -p 8080:8080 my-ubuntu-image

DOCKER COMPOSE

1. **What is Docker Compose?**

   Docker Compose is a tool that helps define and manage multi-container Docker applications. It allows you to use a YAML file (docker-compose.yml) to define services, networks, and volumes for your application. With a single command, docker-compose up, Docker Compose can build, start, and manage multiple containers, making it easier to orchestrate multi-container environments.
   - **Features of Docker Compose**
     1. **Multi-Container Management:**

        Docker Compose allows you to define and run multiple containers at once as a single service or application
     2. **Simplified control:**

        Docker Compose allows you to define and manage multi-container applications in a single YAML file. This simplifies the complex task of orchestrating and coordinating various services, making it easier to manage and replicate your application environment**.**