Technical Test. Tamuka Nongera

Setting up Virtual Machines (Ubuntu)

Step 1: Virtual Box

1. Download the ubuntu ISO file from the Ubuntu website ([Download Ubuntu Desktop | Ubuntu](https://ubuntu.com/download/desktop))
2. Download Virtual Box ([Downloads – Oracle VM VirtualBox](https://www.virtualbox.org/wiki/Downloads))
3. Create new Virtual Machine
   * Click New to start Set up process
   * Name. Enter name of Virtual Machine
   * Folder. Chose file directory to store Virtual Machines
   * ISO Image. Navigate to file directory containing downloaded Ubuntu ISO file.
4. Click Next
   * Fill in user details (Username, Password)
5. Configure Hardware Settings
   * Allocate Base Memory (RAM: 4GB)
   * Processors: Allocate 2 or more.
6. Configure Hard disk
   * Allocate disk Space
7. Finish

Step 2: Installing Microk8s

1. Open Terminal and update package list
   * Sudo apt update
   * Sudo apt update -y
2. Install Snapd (usually a default install, run code to confirm)
   * Sudo apt install snapd -y
3. Install Microk8s using snap command:
   * Sudo snap install microk8s -classic
4. Enable Commonly used Add-ons
   * microk8s enable dns dashboard storage
   * microk8s enable ingress metrics-server helm3
5. Status of MicroK8s
   * microk8s status --wait-ready

Step 3: Installing Docker

1. Install required packages
   * sudo apt install apt-transport-https ca-certificates curl software-properties-common -y
2. Add Docker’s Official GPG Key
   * curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –
3. Add Docker Repository
   * sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable"
4. Update packages
   * sudo apt update
5. Install Docker CE
   * sudo apt install docker-ce -y
6. Use code to verify Docker installation
   * sudo systemctl status docker

Step 4: Install Docker Compose

1. Download Docker Compose
   * sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
2. Apply permissions
   * sudo chmod +x /usr/local/bin/docker-compose
3. Verify Docker Compose Installation:
   * docker-compose –version

Step 5: Install Temporal.io

1. Clone temporal Docker repository
   * git clone https://github.com/temporalio/docker-compose.git
   * cd docker-compose
2. Run Docker Compose
   * docker-compose up

Step 6: Install Temporal Go SDK

1. Install Go
   * sudo apt update
   * sudo apt install golang-go -y
2. Install Temporal SDK
   * go get -u go.temporal.io/sdk

Step 7: Install Java

1. Install OpenJDK
   * sudo apt install openjdk-11-jdk -y
2. Verify the installation
   * java -version

Step 8: Install Neo4j

1. Add Neo4j GPG Key
   * wget -O - https://debian.neo4j.com/neotechnology.gpg.key | sudo apt-key add –
2. Add Neo4j repository
   * echo 'deb https://debian.neo4j.com stable 4.x' | sudo tee /etc/apt/sources.list.d/neo4j.list
3. Install Neo4j
   * sudo apt install neo4j -y
4. Enable Neo4j to start when booting
   * sudo systemctl enable neo4j
5. Check the status of Neo4j
   * sudo systemctl status neo4j

Setting Up SSH Tunnelling

Step 1: Installing SSH

1. Installing SSH server
   * sudo apt install openssh-server -y
2. Enabling SSH services
   * sudo systemctl start ssh
   * sudo systemctl enable ssh
3. IP address of Virtual Machine
   * ip a

Step 2: Create SSH key pair

1. SSHKey
   * ssh-keygen -t rsa -b 2048
2. Copy public key
   * ssh-copy-id user@cluster2\_ip
3. SSH Tunnel from Cluster 1 to Cluster 2
   * ssh -L 7687:localhost:7687 user@cluster2\_ip -N &