**1. We need 6 Ubuntu  with the following resources:**

* 1. 1 Instance with 1CPU, 2RAM, 8 GB Disk - Management t2.small
* 2. 5 Instances with 2CPU, 8GB Ram, Disk Size 20GB 2-worker, 3-master - t2.large

2. Management მანქანაზე შექმენით Public Key (ssh-keygen ბრძანებით)  რომელიც განთავსდება cat /home/ubuntu/.ssh/id\_rsa.pub  ფაილში,  შემდეგ ეს ke

y ჩაამატეთ Master და Worker-ზე შემდეგ მისამართზე vi /home/ubuntu/.ssh/authorized\_keys

3. გახსენით  შემავალი წვდომა ვორქერებისთის და მასტერებისთის Inbound rule all traffic**- Screen**

**1)Install Docker On Management Master and Worker Instance**

* 1. wget https://releases.rancher.com/install-docker/20.10.sh
* 2. sh 20.10.sh
* 3. sudo apt  install docker.io
* 4. sudo systemctl start docker
* 5. sudo systemctl enable docker
* 6. sudo systemctl status docker**- Screen**

**2) Add Docker Group And user On Management Master and Worker Instance**

* 1. sudo addgroup wheel
* 2. sudo usermod -aG wheel ubuntu
* 3. sudo usermod -aG docker ubuntu
* 4. sudo chmod 777 /var/run/docker.sock

**Install Kubectl on Management:**

1. For Linux: <https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/>

* 1. Download the latest release with the command:
* curl -LO "<https://dl.k8s.io/release/$(curl> -L -s <https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl>"
* 2. Validate the binary (optional) Download the kubectl checksum file:
* curl -LO "<https://dl.k8s.io/$(curl> -L -s <https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256>"
* 3. Validate the kubectl binary against the checksum file:
* echo "$(cat kubectl.sha256)  kubectl" | sha256sum --check
* 4. Install kubectl
* sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
* 5. Test to ensure the version you installed is up-to-date:
* kubectl version --client  **- Screen**

**Install Helm on Management, we need for install rancher API:**

* 1. curl -fsSL -o get\_helm.sh <https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3>
* 2. chmod 700 get\_helm.sh
* 3. ./get\_helm.sh **- Screen**

**Install And Configure RKE  on Management**

* **1. # For Linux**
* curl -s <https://api.github.com/repos/rancher/rke/releases/latest> | grep download\_url | grep amd64 | cut -d '"' -f 4 | wget -qi -
* chmod +x rke\_linux-amd64
* sudo mv rke\_linux-amd64 /usr/local/bin/rke
* rke --version**- Screen**
* 2.
* *[+] Cluster Level SSH Private Key Path [~/.ssh/id\_rsa]:*
* *[+] Number of Hosts [1]: 5*
* [+] SSH Address of host (1) [none]: Master Instance 1-2-3 IP
* *[+] SSH Port of host (1) [22]: 22*
* *[+] SSH Private Key Path of host (master IP) [none]:*
* *[-] You have entered empty SSH key path, trying fetch from SSH key parameter*
* *[+] SSH Private Key of host (master IP) [none]:*
* *[-] You have entered empty SSH key, defaulting to cluster level SSH key: ~/.ssh/id\_rsa*
* *[+] SSH User of host (master IP) [ubuntu]:*
* *[+] Is host (master IP a Control Plane host (y/n)? [y]: y*
* *[+] Is host (master IP) a Worker host (y/n)? [n]: n*
* *[+] Is host (master IP) an etcd host (y/n)? [n]: y*
* *[+] Override Hostname of host (master IP) [none]:*
* *[+] Internal IP of host (master IP) [none]:*
* *[+] Docker socket path on host (master IP) [/var/run/docker.sock]:*
* [+] SSH Address of host (2) [none]: Worker Instance 1-2 IP
* *[+] SSH Port of host (2) [22]:*
* *[+] SSH Private Key Path of host () [none]:*
* *[-] You have entered empty SSH key path, trying fetch from SSH key parameter*
* *[+] SSH Private Key of host () [none]:*
* *[-] You have entered empty SSH key, defaulting to cluster level SSH key: ~/.ssh/id\_rsa*
* *[+] SSH User of host () [ubuntu]:*
* *[+] Is host () a Control Plane host (y/n)? [y]: n*
* *[+] Is host () a Worker host (y/n)? [n]: y*
* *[+] Is host () an etcd host (y/n)? [n]: n*
* *[+] Override Hostname of host () [none]:*
* *[+] Internal IP of host () [none]:*
* *[+] Docker socket path on host () [/var/run/docker.sock]:*
* *[+] Network Plugin Type (flannel, calico, weave, canal, aci) [canal]:*
* *[+] Authentication Strategy [x509]:*
* *[+] Authorization Mode (rbac, none) [rbac]:*
* *[+] Kubernetes Docker image [rancher/hyperkube:v1.24.4-rancher1]:*
* *[+] Cluster domain [cluster.local]:*
* *[+] Service Cluster IP Range [10.43.0.0/16]:*
* *[+] Enable PodSecurityPolicy [n]:*
* *[+] Cluster Network CIDR [10.42.0.0/16]:*
* *[+] Cluster DNS Service IP [10.43.0.10]:*
* *[+] Add addon manifest URLs or YAML files [no]:*
* 4. rke up**- Screen**

**Export KubeConfig on Management**

* 1. export KUBECONFIG=/home/ubuntu/kube\_config\_cluster.yml
* 2. echo $KUBECONFIG  **- Screen**

**Generate Certs and install rancher api**

* 1. **DOMAIN=rancher.btu.eu**

**curl**[**https://gist.githubusercontent.com/superseb/b2c1d6c9baa32609a49ee117a27bc700/raw/7cb196e974e13b213ac6ec3105971dd5e21e4c66/selfsignedcert.sh**](https://gist.githubusercontent.com/superseb/b2c1d6c9baa32609a49ee117a27bc700/raw/7cb196e974e13b213ac6ec3105971dd5e21e4c66/selfsignedcert.sh)**| bash -s -- $DOMAIN**

Error: docker: Got permission denied while trying to connect to the Docker daemon socket at unix:

Solution:  sudo chmod 777 /var/run/docker.sock

* 2. **DOMAIN=rancher.btu.eu**
* **helm repo add rancher-latest**[**https://releases.rancher.com/server-charts/latest**](https://releases.rancher.com/server-charts/latest)
* **kubectl create namespace cattle-system**
* **helm install rancher rancher-latest/rancher --namespace cattle-system --set hostname=rancher.btu.eu --set ingress.tls.source=secret --set privateCA=true --set bootstrapPassword=admin --set replicas=1**
* **kubectl -n cattle-system create secret tls tls-rancher-ingress --cert=certs/cert.pem --key=certs/key.pem**
* **sudo cp certs/ca.pem certs/cacerts.pem**

**kubectl -n cattle-system create secret generic tls-ca --from-file=certs/cacerts.pem**

* *3. If rancher pods can't sync change version to fix it!*
* *1. Make a backup of catalogtemplates/system-library-rancher-monitoring ressource*
* *kubectl get catalogtemplates system-library-rancher-monitoring -n cattle-global-data -o yaml > system-library-rancher-monitoring.yaml*
* *2.  Edit the catalogtemplates/system-library-rancher-monitoring ressource*
* *kubectl edit catalogtemplates system-library-rancher-monitoring -n cattle-global-data*
* *3. In the first list item under "spec.versions" edit the "kubeVersion: < 1.22.0-0" to something that matches your kubernetes version. We have set "kubeVersion: '>=1.21.0-0'*
* *4. Restart Docker on Worker and wait 2-3 minutes*
* check kubectl logs -f podName -n NamespaceName
* 4. Write worker IP and rancher On local hostname in /etc/hosts
* 1. *workerHostName*
* 5. Open Rancher UI in browser**-- Screen**
* 6. Reset Password and log in to rancher UI
* kubectl -n cattle-system exec $(kubectl -n cattle-system get pods -l app=rancher | grep '1/1' | head -1 | awk '{ print $1 }') -- reset-password