|  |  |
| --- | --- |
| **DOCUMENT RULES:** | |
| **Task Number / Name:** | **Task 1 / Local Kubernetes setup using minikube** |
| **Task name & column name should be written:** | **Bold (CTRL+B)** |
| **Commands should be written in the after # sign:** | *Italic (CTRL+I) #hostname* |
| **Output photo should be cropped or compressed:**  **Photo could be more than one:**  **If you need extra lines, add the line next after it:** | ***Description photo should be with title bar (CTRL + I + B)*** |
| **All other text should be written:** | Standard |
| **Font name and text size:** | Calibri and 9 |
| **Group name:** | Dev\_ops\_1 |
| **Student name and surname:** | Murad Nesirli |
| **E-mail:** | [muradnasiroof@gmail.com](mailto:terlan_devops_h@mail.ru) |
| **WhatsApp number:** | **+994703959383** |

|  |  |
| --- | --- |
| **Task names** | **Command steps and outputs** |
| 1. **Installation and configuration of Desktop Paravirtualization software on your Desktop:** 2. **Minimal installation of CentOS 7 or Ubuntu on the paravirtual software:** | **Info**: VMware Workstation is Installed with default configuration  **Info**: Guest OS version is Ubuntu 64 bit  **Info**: Installation media is mini-Ubuntu 18.04 Bionic Beaver.iso  ***Please, learn about an OS in lecture first.*** |
| 1. **Test internet access in the cli of Guest VM:** | *#ping 8.8.8.8* |
| 1. **Update** | #sudo apt-get update |
| 1. **and upgrade latest version:** | *#sudo apt-get upgrade* |
| 1. **Check status of firewall and take screenshot of the cli output:** | **For instance: start**, **stop**, **enable**, **disable, status**  *#* *firewall-cmd –state*    Info: the minimal installation is not installed firewall by default |
| 1. **Check Guest OS IP address** | # ip addr |
| 1. **Trouble shooting base command Ping the Guest VM from your desktop using CMD** | *#Ping 192.168.234.143* |
|  | ***Install*** *#Sudo apt install openssh-server* |
| 1. **Trouble shooting base command check SSH port status:** | *#Telnet 192.168.234.143 22*    *Hit enter please*    *If command is not worked, please use appwiz.cpl in your desktop.* |
| 1. **Go to run in your windows desktop:** | *Appwiz.cpl*    Turn on windows fetures on or off -> tick the Telnet Client check box to be on |
| 1. **Take snapshoot of the Guest VM** | Take snapshoot -> Give a name = “Fresh Installation “    As you know there some limitations.  You can delete your unneeded snapshoot afterward. |
| **General Notes** | **Follow the steps**  **Installation kubectl**  **Use docker** |
| 1. **Download minikube, then click copy command and install**   Installation 1 | <https://github.com/kubernetes/minikube>  See the [Getting Started Guide](https://minikube.sigs.k8s.io/docs/start/)  <https://minikube.sigs.k8s.io/docs/start/>    Use commands below to install **minikube**    *#curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64*  *#sudo install minikube-linux-amd64 /usr/local/bin/minikube* |
|  | If command not successful, try to install curl *#sudo apt install curl* -> hit enter button -> *y* -> hit enter then try previous command again to download **minikube** |
| Start your cluster 2 | *#minikube start* -> hit enter -> If you fail look at below  Note 1: try to install container driver and go to driver page. (CTRL + Click <https://minikube.sigs.k8s.io/docs/drivers/>)  Note 2: From a terminal with administrator access (but not logged in as root), run:    Note 3: If **minikube** fails to start, see the for help setting up a **compatible container** or **virtual-machine manager**. We must choose **container “Docker”** because, it is our scope learning.  Go **Docker** link <https://minikube.sigs.k8s.io/docs/drivers/docker/>  **Requirement**  - install docker using this command *#docker pull ubuntu* <https://hub.docker.com/search?q=&type=edition&offering=community&sort=updated_at&order=desc>  - amd64 or arm64  - if using WSL complete these steps first  **Usage**  Start a cluster using the docker driver:  *#minikube start --driver=docker*      To make docker the default driver:  *#minikube config set driver docker*    Then follow the commands below  Then run to deletes images #*minikube delete*    and then run to make images up # *minikube start*  then run to see kube-systems’ running images list:*# minikube kubectl -- get pods -A*  **----------------note-----------------------------------------------**  **Special features**  - no hypervisor required when run on linux  **Known issues**  **Troubleshooting** Verify Docker container type is Linux *#docker info --format '{{.OSType}}'*    **Run with logs**  *#minikube options --alsologtostderr -v=1*    ***-------additional commands---------------------------------***  *#docker info* all about docker    **-------permission—add current user to docker grp-------------**  if you fail you need to set permission the current user: *#sudo usermod -aG docker $USER && newgrp docker* -> hit enter |
| Interact with your cluster 3 |  |
| **3.1** | *#kubectl get po -A* |
| **3.2** | *#minikube kubectl -- get po -A* |
| **3.3** | *#minikube dashboard* |
| **Install kubectl** | Please read all about Kubernetes  [*https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/*](https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/)   1. Install   *#curl -LO* [*https://dl.k8s.io/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256*](https://dl.k8s.io/$(curl%20-L%20-s%20https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256)     1. Validate the binary (optional)   *#curl -LO* [*https://dl.k8s.io/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256*](https://dl.k8s.io/$(curl%20-L%20-s%20https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256)   1. Validate the kubectl binary against the checksum file:   *#echo "$(cat kubectl.sha256) kubectl" | sha256sum –check*   1. If valid, the output is:   *#kubectl: OK*   1. If the check fails, sha256 exits with nonzero status and prints output similar to:   *kubectl: FAILED*     1. sha256sum: WARNING: 1 computed checksum did NOT match 2. *sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl*     Note: If you do not have root access on the target system, you can still install kubectl to the ~/.local/bin directory:   1. #chmod +x kubectl      1. #mkdir -p ~/.local/bin   -----“”””””””ERROR NOTHING HAPPENED TM   1. #mv ./kubectl ~/.local/bin/kubectl     Comment: # and then append (or prepend) ~/.local/bin to $PATH   1. Test to ensure the version you installed is up-to-date: *#kubectl version –client* Or use this for detailed view of version: *#kubectl version --client --output=yaml* |
| **Now 3. (1-3) solved** | *#kubectl get po –A ERROR TM*  *#minikube kubectl -- get po –A ERROR TM*  *#minikube dashboard ------“”””””” MAYBE only THIS SOLVED I DON’T KNOW TM*    *--------“””””””” NOTHING SOLVED ERROR HAPPENED TM* |
| Deploy applications 4 | *#kubectl create deployment hello-minikube --image=k8s.gcr.io/echoserver:1.4*  *----------“””””””” ERROR COMMAND NOT FOUND TM*  *#kubectl expose deployment hello-minikube --type=NodePort --port=8080*  *----------“”””””””” AGAIN ERROR COMMAND NOT FOUND TM*  *#kubectl get services hello-minikube ERROR TM*  *#minikube service hello-minikube JUST NOTHING TM*  *#kubectl port-forward service/hello-minikube 7080:8080 ERROR TM*  Test hello service from locally  <http://localhost:7080/>  #kubectl create deployment balanced --image=k8s.gcr.io/echoserver:1.4    #kubectl expose deployment balanced --type=LoadBalancer --port=8080    #minikube tunnel    #kubectl get services balanced |
|  | *Now you can manage Kubernetes*  *Please learn basics commands*  [*https://minikube.sigs.k8s.io/docs/handbook/controls/*](https://minikube.sigs.k8s.io/docs/handbook/controls/) |
| **Manage your cluster 5** | Pause Kubernetes without impacting deployed applications:  *#minikube pause*    Unpause a paused instance:  *#minikube unpause*    Halt the cluster:  *#minikube stop*    Increase the default memory limit (requires a restart):  *#minikube config set memory 16384*    Browse the catalog of easily installed Kubernetes services:  *#minikube addons list*    Create a second cluster running an older Kubernetes release:  *#minikube start -p aged --kubernetes-version=v1.16.1*    Delete all of the minikube clusters:  *#minikube delete –all* |
| **Take the next step** | * [The minikube handbook](https://minikube.sigs.k8s.io/docs/handbook/) * [Community-contributed tutorials](https://minikube.sigs.k8s.io/docs/tutorials/) * [minikube command reference](https://minikube.sigs.k8s.io/docs/commands/) * [Contributors guide](https://minikube.sigs.k8s.io/docs/contrib/) * Take our [fast 5-question survey](https://forms.gle/Gg3hG5ZySw8c1C24A) to share your thoughts 🙏 |
| **Addtirional command for students’ attention** | #minikube version check installed version of minikube  #minikube start --alsologtostderr -v=7 to debug errors and crashes    to use minikube kubectl to execute kubectl commands that don't require a --namespace flag:  Test if minikube installed or not #minikube kubectl get nodes    #minikube update-check    #minikube start --kubernetes-version=latest    Minikube config files # cat ~/.kube/config |
| **Management tool for Kubernetes cluster is via CLİ** | *#kubectl get pods –A ERROR TM*  *#kubectl get* command lists all pods running ERROR TM  *- A* argument will show you all pods from all namespaces |
| **Minikube web dashboard access** | it will show you link of the Kubernetes  #*minikube dashboard –url*    *#minikube dashboard* if you want break the command press CTRL+C |

|  |  |
| --- | --- |
| **DOCUMENT RULES:** | |
| **Task number / name:** | **Task 2 / Installing Kubernetes using the Docker Client** |
| **Task name & column name should be written:** | **Bold (CTRL+B)** |
| **Commands should be written in the after # sign:** | *Italic (CTRL+I) #hostname* |
| **Output photo should be cropped or compressed:**  **Photo could be more than one:**  **If you need extra lines, add the line next after it:** | ***Description photo should be with title bar (CTRL + I + B)***  Text  Description automatically generated |
| **All other text should be written:** | Standard |
| **Font name and text size:** | Calibri and 9 |
| **Group name:** | Dev\_ops\_1 |
| **Student name and surname:** | Murad Nesirli |
| **E-mail:** | muradnasiroof@gmail.com |
| **WhatsApp number:** | +994703959383 |

|  |  |
| --- | --- |
| **Task names** | **Command steps and outputs** |
| **Installing Kubernetes using the Docker Client:** | Newer edition of Docker allows you install Kubernetes to you PC/laptop.  Docker CE -Community Edition for windows suitable for you  Download stable version or edge version  From this link you can download  <https://docs.docker.com/desktop/windows/install/> NOT INSTALLING FOR WINDOWS VERSION |
| **Install and configure it:** | Note: There is no kubernetes menu on stabile version please Install edge version.    1. There is Kubernetes menu please, enable it and apply.  2. Wait 1 minute, kubernetes will run on this machine  For Starting installation process please follow link below  <https://docs.docker.com/desktop/windows/install/#install-docker-desktop-on-windows> |
| **User manual:** | <https://docs.docker.com/desktop/windows/> |
| **You can use:** |  |
| Usefull commands | *#Kubectl get nodes*    *#Kubectl config get-contexts COMMAND NOT FOUND TM*  *#Kubectl config get-context NAME* Now switched to context “NAME” *COMMAND NOT FOUND TM*  Now you can run some servers  *#Kubectl run hello-kubernetes –-image=k8s.gcr.io/echoserver:1.4 –-port=8080 -> hit enter COMMAND NOT FOUND TM*  Deployment “hello-kubernetes” created  #kubectl expose deployment hello-kubernetes –type=NodePort  service “hello-kubernetes” exposed  #kubectl get service hello-kubernetes ERROR TM  This command for seeing services and you may see browsers  -> localhost:31453 -> enter you will see response  This your first app |
|  |  |