Install Kubernetes and deploy a Selenium Grid cluster on a Kubernetes cluster

Pre-req: docker installed + chmod docker

These commands are part of a set of instructions to install Kubernetes and deploy a Selenium Grid cluster on a Kubernetes cluster. Let's break down what each command does:

1. **sudo apt-get update && sudo apt-get install -y apt-transport-https curl**
    - This command updates the package list on the system (**apt-get update**) and installs the **apt-transport-https** and **curl** packages using **apt-get**. These packages are necessary for later steps.

2. **curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add**
    - This command downloads Google's public key for APT package manager (**apt-key.gpg**) and adds it to the keyring to verify the authenticity of Kubernetes packages.

3. **echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee -a /etc/apt/sources.list.d/kubernetes.list**
    - This command adds the Kubernetes APT repository to the list of package sources by appending the specified line to **/etc/apt/sources.list.d/kubernetes.list.**

4. **sudo apt-get update**
    - This updates the package list again to include the Kubernetes repository.

5. **sudo apt-get install -y kubelet kubeadm kubectl**
    - This installs Kubernetes components: **kubelet**, **kubeadm**, and **kubectl**. **kubelet** is responsible for running containers, **kubeadm** is a tool for initializing a Kubernetes cluster, and **kubectl** is the command-line utility for interacting with Kubernetes clusters.

6. **sudo kubeadm init --pod-network-cidr=10.244.0.0/16**
    - This initializes the Kubernetes master node using **kubeadm** with a specified Pod network CIDR range.

7. **mkdir -p $HOME/.kube**
    - This creates a directory called **.kube** in the user's home directory if it doesn't exist already.

8. **sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config:**
    - This copies the Kubernetes cluster configuration from **/etc/kubernetes/admin.conf** to the user's **.kube/config** file.

9. **sudo chown $(id -u):$(id -g) $HOME/.kube/config:**
    - This changes the ownership of the copied configuration file to the current user.

10. **kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml:**
    - This deploys the Flannel network plugin, which is used for network communication between pods in the Kubernetes cluster.

11. **vi selenium-hub-deployment.yaml**
    - This opens the **selenium-hub-deployment.yaml** file in the Vi text editor for editing. You are expected to make changes or configurations in this file.

12. **kubectl apply -f selenium-hub-deployment.yaml**
    - This deploys the Selenium Hub based on the configuration defined in **selenium-hub-deployment.yaml to** the Kubernetes cluster.

13. **vi selenium-hub-svc.yaml:**
    - This opens the **selenium-hub-svc.yaml** file in the Vi text editor for editing. You are expected to make changes or configurations in this file.

14. **kubectl apply -f selenium-hub-svc.yaml**
    - This deploys a Kubernetes service based on the configuration defined in **selenium-hub-svc.yaml**. This service will expose the Selenium Hub to the cluster.

15. **kubectl get pods -l app=selenium-hub**
    - This command retrieves the pods in the Kubernetes cluster labeled with **app=selenium-hub.** It's used to check if the Selenium Hub pod is running.

16. After confirming that the Selenium Hub is running, it suggests opening a web browser to access the hub at a specific URL **(http://{ip}:32000/ui#).**

17. **vi selenium-node-chrome-deployment.yaml:**
    - This opens the **selenium-node-chrome-deployment.yaml** file in the Vi text editor for editing. You are expected to make changes or configurations in this file.

18. **kubectl apply -f selenium-node-chrome-deployment.yaml**
    - This deploys the Selenium Chrome Node based on the configuration defined in **selenium-node-chrome-deployment.yaml** to the Kubernetes cluster.

19. **kubectl get all**
    - This command retrieves information about all resources in the Kubernetes cluster, including pods, services, deployments, and more.

20. The comment suggests waiting until the pod with the name **selenium-node-chrome** is in a running state.

21. Finally, it advises refreshing the Selenium Grid URL and checking if the Chrome node has been successfully added to the Selenium Grid.

    Please note that some of the steps involve editing YAML configuration files (**selenium-hub-deployment.yaml**, **selenium-hub-svc.yaml**, and **selenium-node-chrome-deployment.yaml**) to customize the deployment according to your needs.
