# Create file shell autorun

Create file shell autorun after restart

|  |
| --- |
| nano /etc/rc.local  sudo chmod +x /etc/rc.local    #!/bin/bash  /usr/local/bin/startup\_script.sh &  exit 0    nano /usr/local/bin/startup\_script.sh  sudo chmod +x startup\_script.sh |

Verify everything works fine:

|  |
| --- |
| sudo /etc/init.d/rc.local start |

Reference: <https://askubuntu.com/questions/9853/how-can-i-make-etc-rc-local-run-on-startup>

Read log

|  |
| --- |
| sudo tail -f /var/log/syslog |

Ensure /etc/rc.local is executable:

|  |
| --- |
| sudo chown root /etc/rc.local sudo chmod 755 /etc/rc.local |

Fix kubelet error

|  |
| --- |
| sudo -i  swapoff -a  exit  strace -eopenat kubectl version    sudo swapoff -a && sudo sed -i '/swap/d' /etc/fstab |

K8s init

|  |
| --- |
| kubectl delete --all all    kubeadm alpha phase kubeconfig user --client-name=kubeadmin --apiserver-advertise-address=192.168.207.129 --kubeconfig-dir=/etc/kubernetes/    kubeadm init phase config --apiserver-advertise-address 192.168.207.129 --cert-dir /etc/kubernetes/pki    kubeadm init --apiserver-advertise-address=192.168.207.129 --cert-dir /etc/kubernetes/pki --pod-network-cidr=10.244.0.0/16 --cert-dir /etc/kubernetes/pki    kubectl get pods -o wide |

Remove k8s taint

|  |
| --- |
| kubectl describe node master-node | grep Taint    kubectl taint nodes --all node-role.kubernetes.io/control-plane:NoSchedule    kubectl taint nodes --all node-role.kubernetes.io/control-plane- |

Apply deployment

|  |
| --- |
| find / -name k8s\_deployment\_service.yaml  sudo kubectl apply -f /var/lib/jenkins/workspace/devsecops-app-num/k8s\_deployment\_service.yaml |

Run ngrok

|  |
| --- |
| ngrok http 8080 |

Docker logs

|  |
| --- |
| docker logs sonarqube -f |

nano /usr/local/bin/startup\_script.sh

|  |
| --- |
| #!/bin/bash  sudo swapoff -a  sudo kubectl version  sudo kubectl get pods -o wide  sudo kubectl taint nodes --all node-role.kubernetes.io/control-plane  sudo kubectl apply -f k8s\_deployment\_service.yaml  #sudo docker stop $(docker ps -a -q)  #sudo docker rm $(docker ps -a -q)  sudo docker start $(docker ps -a -q)  #sudo docker rm $(docker ps -a -q)  #sudo docker run --name sonarqube -d -p 9000:9000 sonarqube:lts  sudo ngrok http 8080    docker run -d --name sonarqube -p 9000:9000 -v sonarqube\_data:/opt/sonarqube/data sonarqube:lts    relay login -k 62a27a59-9d2e-4932-b306-ff42b1ca0089 -s TxZSqnk4yFYC  export RELAY\_KEY=62a27a59-9d2e-4932-b306-ff42b1ca0089 export RELAY\_SECRET=TxZSqnk4yFYC  relay forward --bucket github-jenkins [*http://192.168.207.129:8080/github-webhook/*](http://192.168.207.129:8080/github-webhook/) |