**Pods**

1. kubectl run kuard --image=gcr.io/kuar-demo/kuard-amd64:1
   1. create a pod
2. kubectl get pods
   1. get the status of the pods
3. kubectl delete pod kuard
   1. delete the pod
4. Create a manifest yaml file
   1. Shared
5. Load the Yaml
   1. kubectl apply -f kuard-pod.yaml
6. repeat step 2
7. kubectl describe pods kuard
   1. get more information about the pod created
8. repeat step 4-7 using Jenkins image
9. delete the pods using
   1. kubectl delete pods/kuard
   2. kubectl delete -f jenkins-pod.yaml
10. kubectl port-forward kuard 8081:8080
11. perform same operation on Jenkins
12. access both services from the browser
    1. <http://localhost:8081>
13. kubectl logs kuard
14. try same with Jenkins. Use the information received to log into Jenkins
15. kubectl exec kuard date
16. kubectl exec -it kuard ash
17. perform 15 & 16 with Jenkins.
    1. Note, Jenkins does not have ash, use sh instead
18. For knowledge only, do not use
    1. kubectl cp <pod-name>:/captures/capture3.txt ./capture3.txt
    2. kubectl cp $HOME/config.txt <pod-name>:/config.txt
19. drop both the pods and create a pod with kuard-pod-health.yaml
    1. kubectl apply -f kuard-pod-health.yaml
    2. $ kubectl port-forward kuard 8081:8080
20. Log into the pod using url on step 12
21. Click the “Liveness Probe” tab. You should see a table that lists all of the probes that this instance of kuard has received.
22. Create pods with the ‘kuard-pod-resreq.yaml’, ‘kuard-pod-reslim.yaml’ and ‘kuard-pod-vol.yaml’
    1. Process explained above