Using Static Pods in Kubernetes.

Static pods are a great way to run a pod on a single node without the involvement of the Kubernetes control plane. In this lab, you will have a chance to exercise your knowledge of static pods by creating them in an existing cluster.

The company has built a special diagnostic tool for its K8s nodes. This tool can be run as a container in a K8s Pod, and it collects detailed diagnostic data from the worker node throughout the node lifecycle.

One particularly useful feature is that this tool is able to collect data during the node startup process, before the kubelet begins communicating with the Kubernetes API, or even before a kubelet joins a cluster. To benefit from this, the Pod needs to run without depending on the presence of a Kubernetes API server connection. It will need to be run and managed directly by the kubelet.

Your task is to create a pod to run this diagnostic tool on the Worker Node 1 server. Use the acgorg/beebox-diagnostic:1 image for this Pod.

This lab contains a control plane node and a worker node. And the the static pod is going to be managed entirely by the kubelet on the 1st worker node.

Create a Manifest for a Static Pod

1. Create a static pod manifest file

```
sudo vi /etc/kubernetes/manifests/beebox-diagnostic.yml
```

^ Remember this is on he worker node

We also need to use sudo because the manifest path is going to require root access for us to create a file there.

*The default manifest path for kubelets that are to set up using kubeadm, and the path is going to be etc/Kubernetes/manifests. And then of course at the end input the yml file at the end.

```
apiVersion: v1
kind: Pod
metadata:
  name: beebox-diagnostic
spec:
  containers:
  - name: beebox-diagnostic
    image: acgorg/beebox-diagnostic:1
    ports:
    - containerPort: 80
```

Start Up the Static Pod

So once all that is inputted in the yml file we need to restart the static pod

1. Restart kubelet to start the static pod

```
sudo systemctl restart kublet
```

2. Now in the Control Plane Node we need to check the status of the static pod

```
kubectl get pods
```

Figure 1-1

```
cloud_user@k8s-control:~$ kubectl get pods
NAME                              READY   STATUS    RESTARTS        AGE
beebox-diagnostic-k8s-worker1     1/1     Running   1 (6m45s ago)   6m28s
```

Above you can see on the control node, that the static pod named (worker1) appears here.

We can't actually manage or change the static pod through the Kubernetes API. So we can delete the pod on that name you see in ***Figure 1-1***

3. Now we delete the pod

```
kubectl delete pod beebox-diagnostic-k8s-worker1
```

After deleting this pod and viewing the status it went from pending to running.