**Application Deployment using Static Pod:**

* Static pods will be managed by kubelet daemon on specific nodes. As we have seen till now, we have executed kubectl commands to create a pod on kubernetes cluster, these requests are accepted by the kube API server to do validation, schema check then it will pass the information to the etcd and kube scheduler. Later in the compute plane, kubelet starts creating a pod.
* Static pods will not be controlled by the kube API server (kubectl or API calls), it is only managed by the kubelet daemon which is running on a specific node. When there is a situation to launch a static pod in a particular node then we need to go to that node and specify the kubelet to launch the pod on that exact node.
* Deleting static pods through API or kubectl commands will not affect the application, since kubelet will recreate them. Even though we try to control it with the kubectl, it doesn't work.
* Static pod names will be added with the suffix of node name.
* Static pods manifest can be defined through two approaches. They are File System-hosted or web-hosted.
* File system-hosted approach is to create the yaml or json files to deploy a pod and place them under default directory /etc/kubernetes/manifests which are read by kubelet.
* Kubelet can download manifest files from websites by passing –manifest-url=<URL> argument for web-hosted approach. Therefore, Web-hosted approach is to place yaml or json files as manifest at some location of any browser or a website and when we need to create a static pod, these files are downloaded from that location.

**Flat no:205, 2nd Floor,NILGIRIBlock, Adhitya Enclave, Ameerpet,Hyderabadh-500016,Telangana India** [**support@visualpathtech.com**](mailto:support@visualpathtech.com)**+9190300222850,+919154152888**