Agenda for Dockers & Kubernetes Training – 5 days:
Docker version : 19.03, K8s version: 1.18

Day1:

- Monolith Applications and their limitations, need for Microservice based Architecture, introduction to Cloud Native Application Design & Serverless Architecture, Scalable Deployment environments & container ecosystem
- Container analogy, difference between containers & VMs
- Docker Workflow
- Docker Architecture – Engine, Container, Docker host, Image, Registry
- Docker Installation - Linux Based machine (AWS Cloud)
- Docker 101 commands, attaching & detaching from containers, daemon containers, starting, stopping & removing containers
- Sample exercises for running various application containers – nginx, MySQL, tomcat, MongoDB etc
- Docker logs & port forward, debugging containers by starting docker daemon in Debug mode
- Building Docker Images using Dockerfile, building a simple C++ Docker Image & tagging it

Day 2:

- Dockerfile detailed syntax – FROM, COPY, ADD, ENV, USER, WORKDIR, VOLUME, EXPOSE, RUN, CMD, ENTRYPOINT, .dockerignore file
- Exercise to build real time web application facebooc (C with SQL Lite)Docker Image using two approaches and pushing them to DockerHub Registry
- Building & storing images, pushing them to public repositories, Docker hub registration
- Concept of Private/Enterprise Registries, creating a local private registry and pushing newly built docker image to private registry – both Docker Registry2 and Sonatype Nexus OSS
- Understanding Storage, Layers & inspecting docker images with exercise
- Concept of Docker Volumes, creating multiple docker volumes, sharing volumes across containers, read-only volumes
- Overview of Docker Networking, CNI/CNM concepts
- Overview of Docker Compose & Docker Swarm, exercise to bring up multiple container based application using Docker Compose, perform rolling updates & scaling up of services in Docker Swarm
- Docker Management using Portainer : https://portainer.io , Monitoring using CTOP

Day 3:

- Security best practices while building docker images, using Lints, analyzing Docker images for vulnerabilities etc
- Introduction to Kubernetes, Installation of Minikube & Kubectl on Linux VM
- Basic Kubectl commands – get pods, attach, run, describe pod, pod logs
- Kubernetes Architecture & Components(Kubelet, kube-proxy, etcd, Scheduler etc), overview of Cloud specific K8s implementations – EKS, AKS etc
- Pods, lifecycle of Pods, Single Container Pod v/s multi-container Pods, Sidecar Design Pattern, exercise to set-up multi-container Pod with volume sharing

- Need to Organize Pods, Labels, assigning Pods to specific Nodes using NodeSelector, Pod affinity & anti-affinity, Taints & Tolerations, exercises to set up Pods with some affinity or Toleration, check its status when it does not find any matching node for scheduling/execution
- Namespaces, organize Pods by running them in specific namespaces
- Health Checks & self-healing applications, Difference between Liveness & Readiness Probes, different types of Probes (httpGet, tcpSocket, exec) and arguments (initialDelaySeconds, periodSeconds, failureThreshold etc), exercise to set up probes for Pods and verify the restart in case of failures, best practices for setting up Application Probes

Day 4:

- Services, need for having services, deep dive into types services – ClusterIP, NodePort, LoadBalancer and external (CNI/CNM), exercises to run a pod & expose it as Service, use case where to use NodePort
- Pod Networking, Introduction to Ingress, set up NGINX ingress for configuring two different Pods to different end points
- Volumes, different types of volumes (nfs, hostpath, cloud specific & distributed file system), importance of Volumes in Pods, exercise to set up a Pod with a volume of type emptyDir
- Introduction to PersistentVolume & PersistentVolumeClaim, exercise to define Pod Manifest which will mount a volume based on PVC that it binds to
- ConfigMaps & Secrets, need to inject run time configurations to Pods, exercise to set up ConfigMaps & Secrets and inject them into Pods as VolumeMounts and Environment Variables
- K8s Work load Objects with exercises– ReplicaSet, Deployment, Statefulset, DaemonSet, Job & CronJob

Day 5:

- Deep dive into release management using Deployment, setting up simple Tomcat deployment using RollingUpdate strategy, importance of maxSurge & maxUnavailable parameters, checking status of deployment using rollout and rolling back updates in case of any issues
- StatefulSets (v/s ReplicaSet) and its use, exercise to set up a Statefulset with PVC, govern it with headless service for providing unique DNS (will cover this topic if required)
- Security best practices, Authentication & Authorization, Role Based Access (RBAC), TLS and certificates (examples of Lets Encrypt for signing applications), Network Policy & Security Context (allowPrivelegeEscalation & runAsUser etc), exercise to set up simple network policy for governing access to Pods based on Labels, serviceAccounts and its importance
- Advance Topics – resourceQuotas, assigning Pod resource (CPU & Memory) requests & limits, Pod monitoring using Metrics Server, exercise to check Pod resource usage using Metrics Server, Auto scaling in K8s (HPA & VPA)
- Introduction to Helm Charts, Helm 2 v/s Helm 3,Install Helm 3.0 & deploy a simple Grafana dashboard using Helm Charts
- Monitoring K8s cluster, overview of tools such as Prometheus & Grafana
- Demo to set up production grade K8s cluster using Kubeadm tool
- Summarize & wrap up

Lab Set-Up

- All participants will be provided with AWS EC2 Instances – T2.MEDIUM, one Ubuntu VM per participant, so that all hands-on exercises can be carried out on their individual VMs
- For Docker Swarm, 2 participants will be grouped to form a Docker Swarm and carry out exercises for scaling up services, draining nodes, performing rolling updates in Docker Swarm
- Single Node Kubernetes Cluster Version 1.18.0 will be installed on each participant VM using KubeAdm tool, along with Flannel CNI plugin