Architecture, Installation & Maintenance

Weight : 4

1) For this question, please set this context (In exam, diff cluster name)

```
kubectl config use-context kubernetes-admin@kubernetes
```

Find the Node that consumes the most MEMORY in all cluster(currently we have single cluster). Then, store the result in the file `high_memory_node.txt` with the following format: `current_context,node_name` .

Solution:-

Step 1: kubectl top node
Step 2: echo "kubernetes-admin@kubernetes,controlplane" > high_memory_node.txt

---

Weight : 4

2) For this question, please set this context (In exam, diff cluster name)

```
kubectl config use-context kubernetes-admin@kubernetes
```

`application-pod` pod is running, save All ERROR's pod logs only in `poderrorlogs.txt`

Solution:-

Step 1: kubectl logs application-pod | grep ERROR
Step 2: Save it in file poderrorlogs.txt

Weight : 4

3) For this question, please set this context (In exam, diff cluster name)

```
kubectl config use-context kubernetes-admin@kubernetes
```

`alpine-reader-pod` pod is running, save All INFO and ERROR's pod logs in `podlogs.txt`

Solution:-

Step 1: kubectl logs alpine-reader-pod | grep -E INFO|ERROR
Step 2: Save it in file podlogs.txt

Weight : 4

4) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

`log-reader-pod` pod is running, save All pod logs in `podalllogs.txt`

Solution:-

Step 1: kubectl logs log-reader-pod
Step 2: Save it in file podalllogs.txt

Weight : 2

5) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

Decode the contents of the existing secret named `database-data` in the `database-ns` namespace and save the decoded content into a file located at `decoded.txt`

Solution:-

Step 1: get the encoded value

kubectl get secret database-data -n database-ns -o yaml

Step 2: Decode the encoded value

echo "c2VjcmV0" | base64 -d

Step 3: Save it in file

echo "secret" > decoded.txt

Weight : 2

6) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

Create a Kubernetes Secret named `database-app-secret` in the default namespace using the contents of the file `database-data.txt`

Solution:- kubectl create secret generic database-app-secret --from-file=database-data.txt -n default

---

Weight : 10

7) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

Upgrade `controlplane` node `kubeadm` , `cluster` and `kubelet` to next version.

EXAMPLE: If current version is `v1.27.1` then upgrade to `v1.27.2`

Solution:-

Step 1: Find the latest patch release for Kubernetes 1.27 using the OS package manager:

```
apt update
```

```
apt-cache madison kubeadm
```

Step 2: Upgrade kubeadm:

```
apt-mark unhold kubeadm && \
```

```
apt-get update && apt-get install -y kubeadm=1.27.2-00 && \
```

```
apt-mark hold kubeadm
```

Step 3: Verify that the download works and has the expected version:

```
kubeadm version
```

Step 4: Verify the upgrade plan:

```
kubeadm upgrade plan
```

Step 5: Choose a version to upgrade(cluster) to, and run the appropriate command.

```
sudo kubeadm upgrade apply v1.27.2
```

Step 6: Upgrade the kubelet and kubectl:

```
apt-mark unhold kubelet kubectl && \
```

```
apt-get update && apt-get install -y kubelet=1.27.2-00
kubectl=1.27.2-00 && \
```

```
apt-mark hold kubelet kubectl
```

Step 7: Restart the kubelet:

```
sudo systemctl daemon-reload
```

```
sudo systemctl restart kubelet
```

Weight : 2

8) For this question, please set this context (In exam, diff cluster name)

```
kubectl config use-context kubernetes-admin@kubernetes
```

you have a script named `svc-filter.sh` . Update this script to include a command that filters and displays the value of `target port` of a service named `redis-service` using `jsonpath` only.

Solution:- add this below command in svc-filter.sh file

```
kubectl get svc redis-service -o jsonpath='{.spec.ports[0].targetPort}'
```

OR

```
kubectl get service redis-service -o jsonpath='{.spec.ports[0].targetPort}'
```

Weight : 8

9) For this question, please set this context (In exam, diff cluster name)

```
kubectl config use-context kubernetes-admin@kubernetes
```

Create a Kubernetes Pod configuration to facilitate real-time monitoring of a log file. Specifically, you need to set up a Pod named `alpine-pod-pod` that runs an Alpine Linux container.

Requirements:

- Name the Pod `alpine-pod-pod` .
- Use `alpine:latest` image
- Configure the container to execute the `tail -f /config/log.txt` command using `/bin/sh` to continuously monitor and display the contents of a log file.
- Set up a volume named `config-volume` that maps to a ConfigMap named `log-configmap` , this `log-configmap` already available.
- Ensure the Pod has a restart policy of `Never` .

Solution:-

Step 1: Get the pod template

```
kubectl run alpine-pod-pod --image=alpine:latest --dry-run=client -o yaml > pod.yaml
```

Step 2: Update the pod template

```
apiVersion: v1
kind: Pod
metadata:
  name: alpine-pod-pod
spec:
  containers:
  - name: alpine-container
    image: alpine:latest
    command: ["/bin/sh", "-c"]
    args:
    - "tail -f /config/log.txt"
    volumeMounts:
    - name: config-volume
```

```
      mountPath: /config
  volumes:
  - name: config-volume
    configMap:
      name: log-configmap
  restartPolicy: Never
```

Weight : 2

10) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

you have a script named `pod-filter.sh` . Update this script to include a command that filters and displays the label with the value `application` of a pod named `nginx-pod` using `jsonpath` only.

Solution:- add this below command in pod-filter.sh file

```
kubectl get pod nginx-pod -o=jsonpath='{.metadata.labels.application}'
```

Weight : 4

11) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

You have a service account named `group1-sa` , a ClusterRole named `group1-role-cka` , and a ClusterRoleBinding named `group1-role-binding-cka` . Your task is to update the permissions for the `group1-sa` service account so that it can only `create` , `get` and `list` the `deployment` and no other resources in the cluster.

Solution:

Step 1: run ClusterRole edit command

```
kubectl edit clusterrole group1-role-cka
```

Step 2: Update From-

```
rules:
- apiGroups:
  - apps
  resources:
  - deployments
  verbs:
  - get
```

To -

```
rules:
- apiGroups:
  - apps
  resources:
  - deployments
  verbs:
  - get
  - create
  - list
```

Weight : 4

12) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

Find the pod that consumes the most CPU in all namespace(including kube-system) in all cluster(currently we have single cluster). Then, store the result in the file `high_cpu_pod.txt` with the following format: `pod_name,namespace` .

Solution:- Step 1:Check which pod consumed the most CPU `kubectl top po -A`

Step 2: Save it in file `echo "kube-apiserver-controlplane,kube-system" > high_cpu_pod.txt`

Weight : 4

13) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

`product` pod is running. when you access logs of this pod, it displays the output `Mi Tv Is Good`

Please update the pod definition file to utilize an environment variable with the value `Sony Tv Is Good` Then, recreate this pod with the modified configuration.

Solution:-

Step 1: edit pod

```
kubectl edit pod product
```

Step 2: Update and Save(wq). From-

```
  containers:
  - command:
    - sh
    - -c
    -t echo 'Mi Tv Is Good' && sleep 3600
```

To-

```
  containers:
  - command:
    - sh
```

```
    - -c
    -t echo 'Sony Tv Is Good' && sleep 3600
```

This will give update pod template (Ex: /tmp/kubectl-edit-<random-number>.yaml)

Step 3: To recreate pod(fast use --force flag) with update template

```
kubectl replace -f /tmp/kubectl-edit-2137593717.yaml --force
```

Weight : 10

14) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

`etcd-controlplane` pod is running in `kube-system` environment, take backup and store it in `/opt/cluster_backup.db` file, and also store backup console output store it in `backup.txt`

Solution:-

Step 1: Take backup

```
etcdctl --endpoints=https://127.0.0.1:2379 --cacert=/etc/kubernetes/pki/etcd/ca.crt
--cert=/etc/kubernetes/pki/etcd/server.crt --key=/etc/kubernetes/pki/etcd/server.key snapshot save
/opt/cluster_backup.db
```

Step 2: Save console o/p in a file `backup.txt`

Weight : 10

15) For this question, please set this context (In exam, diff cluster name)

`kubectl config use-context kubernetes-admin@kubernetes`

`etcd-controlplane` pod is running in `kube-system` environment, take backup and store it in `/opt/cluster_backup.db` file.

ETCD backup is stored at the path `/opt/cluster_backup.db` on the `controlplane` node. for --data-dir use `/root/default.etcd` , restore it on the `controlplane` node itself and , and also store restore console output store it in `restore.txt`

Solution:-

Step 1: run restore command

```
etcdctl snapshot restore /opt/cluster_backup.db --data-dir=/root/default.etcd
--cacert=/etc/kubernetes/pki/etcd/ca.crt --cert=/etc/kubernetes/pki/etcd/server.crt
--key=/etc/kubernetes/pki/etcd/server.key
```

Step 2: Save console o/p in a file `restore.txt`