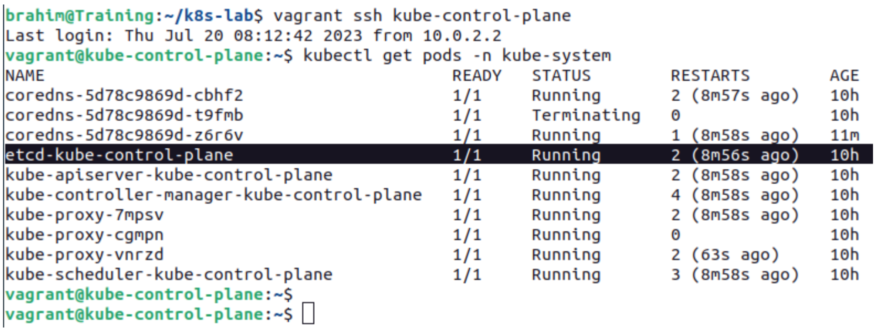
Lab03 – Backuping up & Restoring etcd

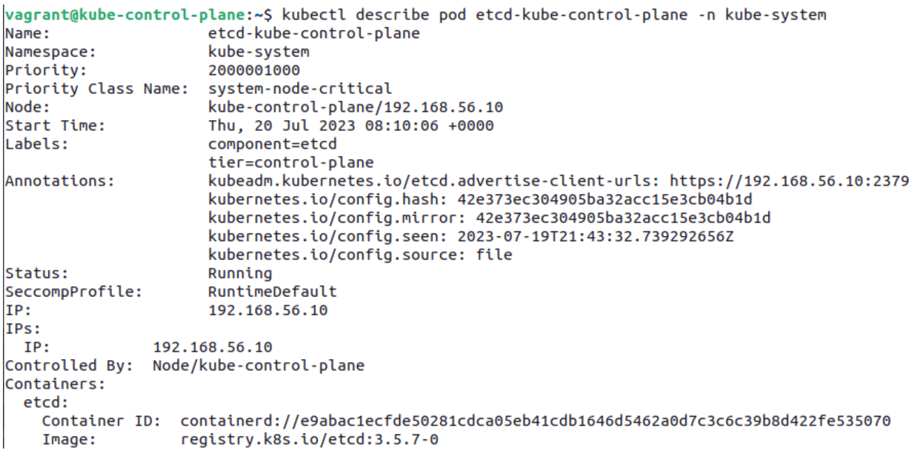
In this exercise, you will identify the configuration of the etcd database, back it up and restore the original database from a backup file. The command line tool `etcdctl` has already been pre-installed on the control plane node.

1. Shell into control plane node and check that all nodes have been correctly registered and are in the "Ready" status.

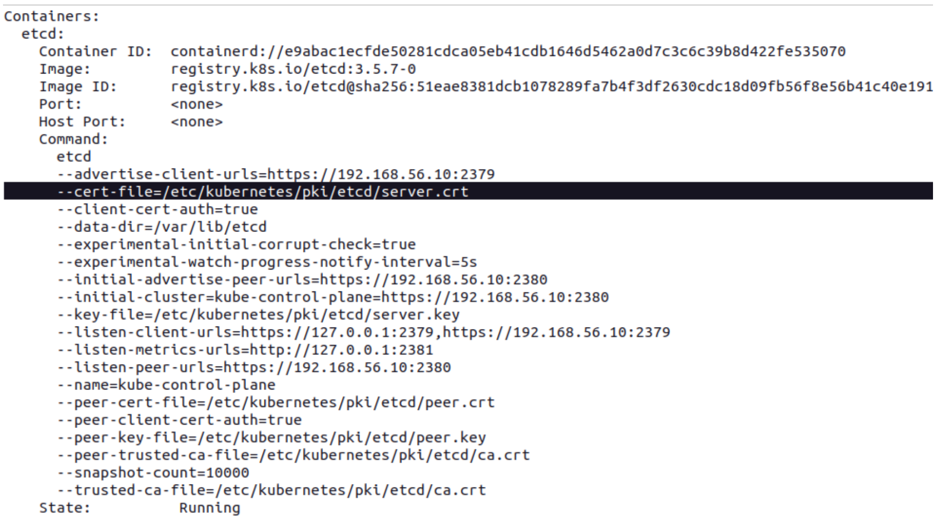


```
brahim@Training:~/k8s-lab$ vagrant ssh kube-control-plane
Last login: Thu Jul 20 08:12:42 2023 from 10.0.2.2
vagrant@kube-control-plane:~$ kubectl get pods -n kube-system
NAME                                         READY   STATUS        RESTARTS        AGE
coredns-5d78c9869d-cbhf2                     1/1     Running       2 (8m57s ago)   10h
coredns-5d78c9869d-t9fmb                     1/1     Terminating   0               10h
coredns-5d78c9869d-z6r6v                     1/1     Running       1 (8m58s ago)   11m
etcd-kube-control-plane                      1/1     Running       2 (8m56s ago)   10h
kube-apiserver-kube-control-plane            1/1     Running       2 (8m58s ago)   10h
kube-controller-manager-kube-control-plane   1/1     Running       4 (8m58s ago)   10h
kube-proxy-7mpsv                             1/1     Running       2 (8m58s ago)   10h
kube-proxy-cgmpn                             1/1     Running       0               10h
kube-proxy-vnrzd                             1/1     Running       2 (63s ago)     10h
kube-scheduler-kube-control-plane            1/1     Running       3 (8m58s ago)   10h
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ 
```

2. Unsing `describe` command, reveals the configuration of the etcd service and look for the values of `--listen-client-urls`, `--cert-file` and `--trusted-ca-file` options.

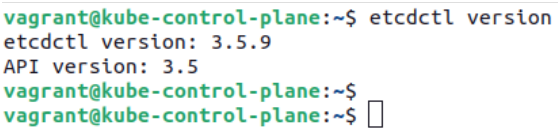


```
vagrant@kube-control-plane:~$ kubectl describe pod etcd-kube-control-plane -n kube-system
Name:                 etcd-kube-control-plane
Namespace:            kube-system
Priority:             2000001000
Priority Class Name:  system-node-critical
Node:                 kube-control-plane/192.168.56.10
Start Time:           Thu, 20 Jul 2023 08:10:06 +0000
Labels:               component=etcd
                      tier=control-plane
Annotations:          kubeadm.kubernetes.io/etcd.advertise-client-urls: https://192.168.56.10:2379
                      kubernetes.io/config.hash: 42e373ec304905ba32acc15e3cb04b1d
                      kubernetes.io/config.mirror: 42e373ec304905ba32acc15e3cb04b1d
                      kubernetes.io/config.seen: 2023-07-19T21:43:32.739292656Z
                      kubernetes.io/config.source: file
Status:               Running
SeccompProfile:       RuntimeDefault
IP:                   192.168.56.10
IPs:
  IP:           192.168.56.10
Controlled By:  Node/kube-control-plane
Containers:
  etcd:
    Container ID:  containerd://e9abac1ecfde50281cdca05eb41cdb1646d5462a0d7c3c6c39b8d422fe535070
    Image:         registry.k8s.io/etcd:3.5.7-0
```



```
Containers:
  etcd:
    Container ID:  containerd://e9abac1ecfde50281cdca05eb41cdb1646d5462a0d7c3c6c39b8d422fe535070
    Image:         registry.k8s.io/etcd:3.5.7-0
    Image ID:      registry.k8s.io/etcd@sha256:51eae8381dcb1078289fa7b4f3df2630cdc18d09fb56f8e56b41c40e191
    Port:          <none>
    Host Port:     <none>
    Command:
      etcd
      --advertise-client-urls=https://192.168.56.10:2379
      --cert-file=/etc/kubernetes/pki/etcd/server.crt
      --client-cert-auth=true
      --data-dir=/var/lib/etcd
      --experimental-initial-corrupt-check=true
      --experimental-watch-progress-notify-interval=5s
      --initial-advertise-peer-urls=https://192.168.56.10:2380
      --initial-cluster=kube-control-plane=https://192.168.56.10:2380
      --key-file=/etc/kubernetes/pki/etcd/server.key
      --listen-client-urls=https://127.0.0.1:2379,https://192.168.56.10:2379
      --listen-metrics-urls=http://127.0.0.1:2381
      --listen-peer-urls=https://192.168.56.10:2380
      --name=kube-control-plane
      --peer-cert-file=/etc/kubernetes/pki/etcd/peer.crt
      --peer-client-cert-auth=true
      --peer-key-file=/etc/kubernetes/pki/etcd/peer.key
      --peer-trusted-ca-file=/etc/kubernetes/pki/etcd/ca.crt
      --snapshot-count=10000
      --trusted-ca-file=/etc/kubernetes/pki/etcd/ca.crt
    State:          Running
```

3. Find out the version of etcd running in the cluster.

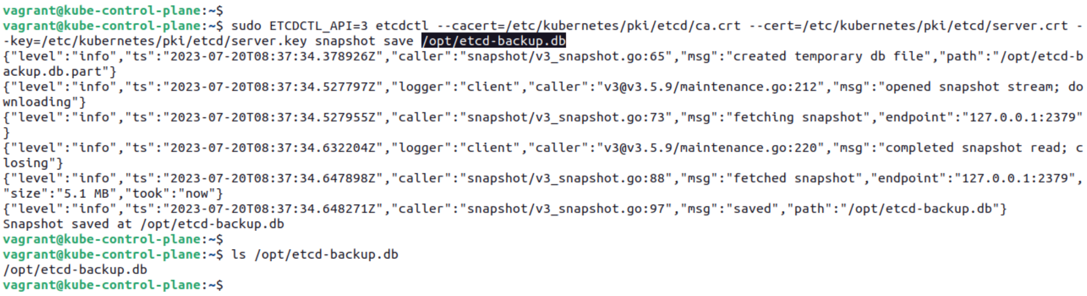


```
vagrant@kube-control-plane:~$ etcdctl version
etcdctl version: 3.5.9
API version: 3.5
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ 
```

4. Use the `etcdctl` command to create the backup with version 3 of the tool.

   For a good starting point, copy the command from https://kubernetes.io/docs/tasks/administer-cluster/configure-upgrade-etcd/#built-in-snapshot link. Provide the mandatory command line options `--cacert`, `--cert`, and `--key`.

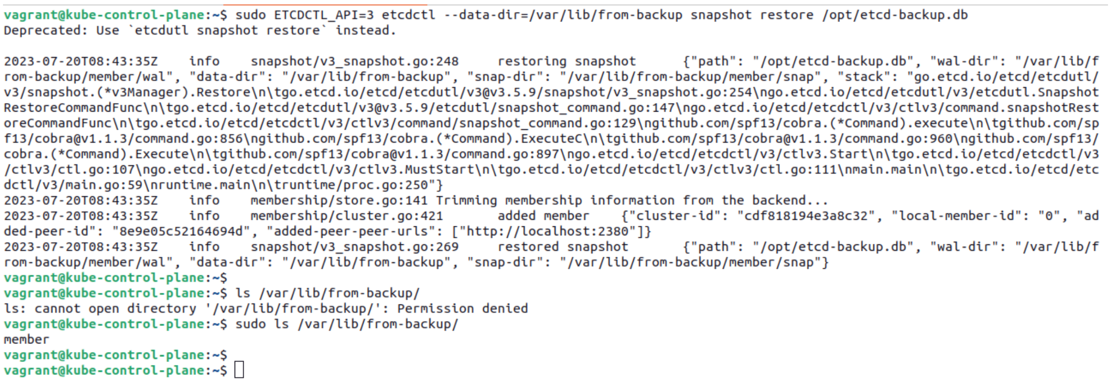
   The option `--endpoints` is not needed as we are running the command on the same server as etcd.



```
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ sudo ETCDCTL_API=3 etcdctl --cacert=/etc/kubernetes/pki/etcd/ca.crt --cert=/etc/kubernetes/pki/etcd/server.crt -
-key=/etc/kubernetes/pki/etcd/server.key snapshot save /opt/etcd-backup.db
{"level":"info","ts":"2023-07-20T08:37:34.378926Z","caller":"snapshot/v3_snapshot.go:65","msg":"created temporary db file","path":"/opt/etcd-b
ackup.db.part"}
{"level":"info","ts":"2023-07-20T08:37:34.527797Z","logger":"client","caller":"v3@v3.5.9/maintenance.go:212","msg":"opened snapshot stream; do
wnloading"}
{"level":"info","ts":"2023-07-20T08:37:34.527955Z","caller":"snapshot/v3_snapshot.go:73","msg":"fetching snapshot","endpoint":"127.0.0.1:2379"
}
{"level":"info","ts":"2023-07-20T08:37:34.632204Z","logger":"client","caller":"v3@v3.5.9/maintenance.go:220","msg":"completed snapshot read; c
losing"}
{"level":"info","ts":"2023-07-20T08:37:34.647898Z","caller":"snapshot/v3_snapshot.go:88","msg":"fetched snapshot","endpoint":"127.0.0.1:2379",
"size":"5.1 MB","took":"now"}
{"level":"info","ts":"2023-07-20T08:37:34.648271Z","caller":"snapshot/v3_snapshot.go:97","msg":"saved","path":"/opt/etcd-backup.db"}
Snapshot saved at /opt/etcd-backup.db
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ ls /opt/etcd-backup.db
/opt/etcd-backup.db
vagrant@kube-control-plane:~$
```

After running the command, the file `/tmp/etcd-backup.db` should been created.

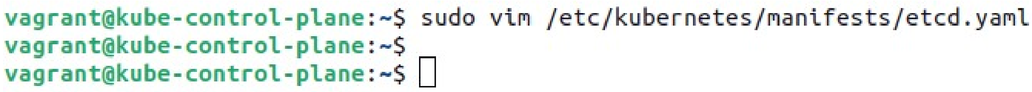
5. Use the `etcdctl` command to restore etcd from the backup. Provide the `--data-dir` command line option to restore data to `/tmp/from-backup` directory.



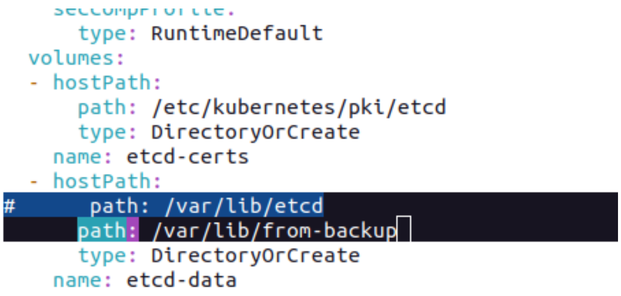
```
vagrant@kube-control-plane:~$ sudo ETCDCTL_API=3 etcdctl --data-dir=/var/lib/from-backup snapshot restore /opt/etcd-backup.db
Deprecated: Use `etcdutl snapshot restore` instead.

2023-07-20T08:43:35Z    info    snapshot/v3_snapshot.go:248     restoring snapshot      {"path": "/opt/etcd-backup.db", "wal-dir": "/var/lib/f
rom-backup/member/wal", "data-dir": "/var/lib/from-backup", "snap-dir": "/var/lib/from-backup/member/snap", "stack": "go.etcd.io/etcd/etcdutl/
v3/snapshot.(*v3Manager).Restore\n\tgo.etcd.io/etcd/etcdutl/v3@v3.5.9/snapshot/v3_snapshot.go:254\ngo.etcd.io/etcd/etcdutl/v3/etcdutl.Snapshot
RestoreCommandFunc\n\tgo.etcd.io/etcd/etcdutl/v3@v3.5.9/etcdutl/snapshot_command.go:147\ngo.etcd.io/etcd/etcdctl/v3/ctlv3/command.snapshotRest
oreCommandFunc\n\tgo.etcd.io/etcd/etcdctl/v3/ctlv3/command/snapshot_command.go:129\ngithub.com/spf13/cobra.(*Command).execute\n\tgithub.com/sp
f13/cobra@v1.1.3/command.go:856\ngithub.com/spf13/cobra.(*Command).ExecuteC\n\tgithub.com/spf13/cobra@v1.1.3/command.go:960\ngithub.com/spf13/
cobra.(*Command).Execute\n\tgithub.com/spf13/cobra@v1.1.3/command.go:897\ngo.etcd.io/etcd/etcdctl/v3/ctlv3.Start\n\tgo.etcd.io/etcd/etcdctl/v3
/ctlv3/ctl.go:107\ngo.etcd.io/etcd/etcdctl/v3/ctlv3.MustStart\n\tgo.etcd.io/etcd/etcdctl/v3/ctlv3/ctl.go:111\nmain.main\n\tgo.etcd.io/etcd/etc
dctl/v3/main.go:59\nruntime.main\n\truntime/proc.go:250"}
2023-07-20T08:43:35Z    info    membership/store.go:141 Trimming membership information from the backend...
2023-07-20T08:43:35Z    info    membership/cluster.go:421       added member    {"cluster-id": "cdf818194e3a8c32", "local-member-id": "0", "ad
ded-peer-id": "8e9e05c52164694d", "added-peer-peer-urls": ["http://localhost:2380"]}
2023-07-20T08:43:35Z    info    snapshot/v3_snapshot.go:269     restored snapshot       {"path": "/opt/etcd-backup.db", "wal-dir": "/var/lib/f
rom-backup/member/wal", "data-dir": "/var/lib/from-backup", "snap-dir": "/var/lib/from-backup/member/snap"}
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ ls /var/lib/from-backup/
ls: cannot open directory '/var/lib/from-backup/': Permission denied
vagrant@kube-control-plane:~$ sudo ls /var/lib/from-backup/
member
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ 
```

After running the command, you should be able to find the restored backup in the directory `/var/lib/from-backup`.

6. Edit the YAML manifest of the etcd Pod which can be found at `/etc/kubernetes/manifests/etcd.yaml`. Change the value of the attribute `spec.volumes.hostPath` with the name `etcd-data` from the original value `/var/lib/etcd` to `/var/lib/from-backup`.



```
vagrant@kube-control-plane:~$ sudo vim /etc/kubernetes/manifests/etcd.yaml
vagrant@kube-control-plane:~$
vagrant@kube-control-plane:~$ 
```



```
    seccompProfile:
      type: RuntimeDefault
  volumes:
  - hostPath:
      path: /etc/kubernetes/pki/etcd
      type: DirectoryOrCreate
    name: etcd-certs
  - hostPath:
#      path: /var/lib/etcd
      path: /var/lib/from-backup
      type: DirectoryOrCreate
    name: etcd-data
```

The `etcd-kube-control-plane` Pod will be recreated and points to the restored backup directory.