Certified Kubernetes Administrator (CKA) — Tips and Tricks — Part 3

Arun Ramakani Follow
Dec 6 · 4 min read

Today let's look into ETCD backup. If you get this question it's a jackpot. You can score the full mark in less than a minute, if you know how to do it. This will save some time for other questions. ETDC back up is a one-line command, but you need to collect a few pieces of information needed to execute the command.

Tip 1: Parts of the ETDC Backup Command

To back up the cluster, we should use the below command

> ETCDCTL_API=3 etcdctl — endpoints=[ENDPOINT] — cacert=[CA CERT] — cert=[ETCD SERVER CERT] — key=[ETCD SERVER KEY] snapshot save [BACKUP FILE NAME]

Executing the command will immediately give feedback if the backup is taken correctly. In case if you have not got the command correctly, you will have immediate feedback of failure.

```
master $ ETCDCTL_API=3 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 /backup/snapshot.db
Snapshot saved at /backup/snapshot.db
master $
master $
```

The above instruction has 6 important parts to it

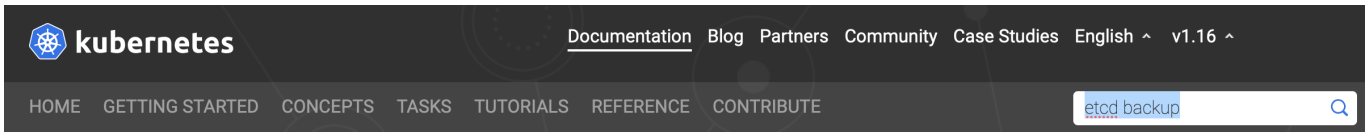
1. Command to take a backup — See **Tip 2** on how to escape memorizing
2. *ENDPOINT* — See **Tip 3** on how to get this value
3. *CA CERT* — See **Tip 3** on how to get this value
4. *ETCD SERVER CERT* — See **Tip 3** on how to get this value
5. *ETCD SERVER KEY* — See **Tip 3** on how to get this value
6. *BACKUP FILE NAME — This will be given as a part of question itself*

Any missing options will throw an error

```
master $ ETCDCTL_API=3 etcdctl --endpoints=https://[127.0.0.1]:2379 --cacert=/etc/kubernetes/pki/etcd/ca.crt --cert=/etc/kubernetes/pki/etcd/server.crt snapshot save /backup/snapshot.db
Error: KeyFile and CertFile must both be present[key: , cert: /etc/kubernetes/pki/etcd/server.crt]
```

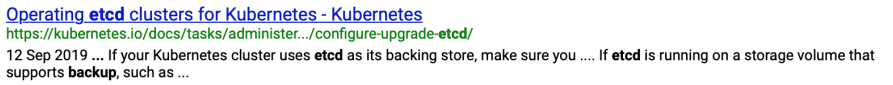
Tips 2: No Need to Memorize the Command

You don't need to memorize the command for backing up ETDC. You will be allowed to refer to the Kubernetes documentation page during the exam. From the Kubernetes documentation page (*doc page*) search for "etcd backup", then from the results click the first link "Operating etcd clusters …".

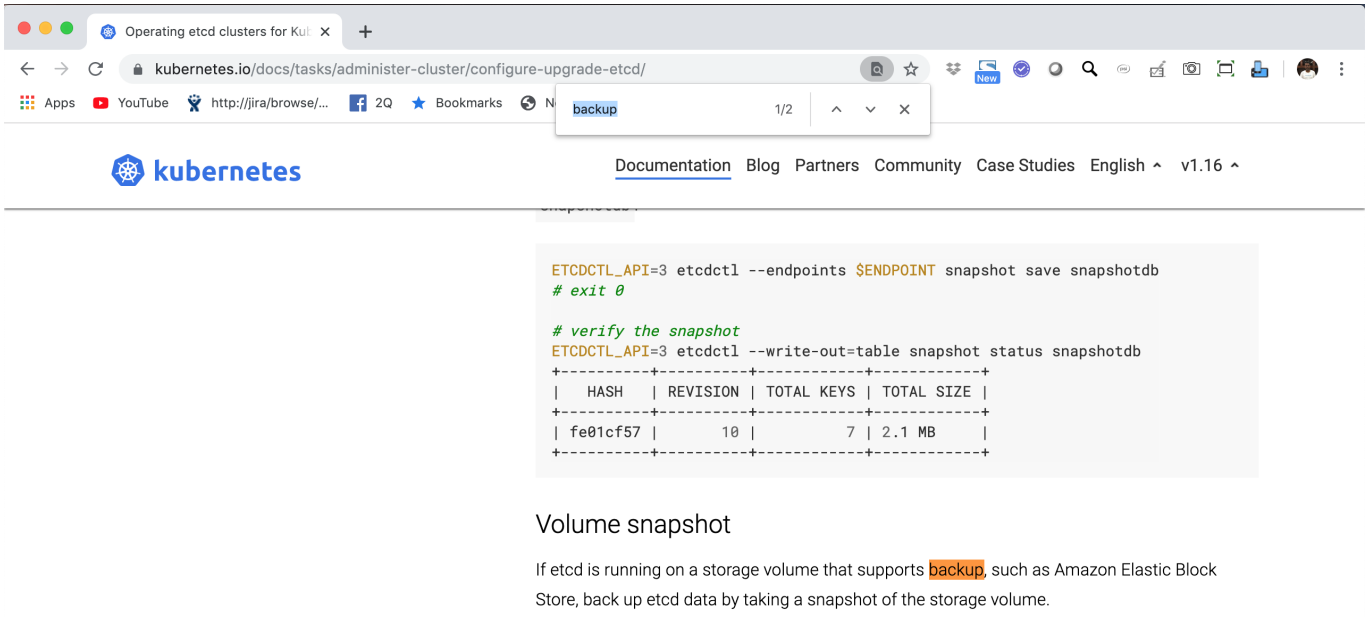


Search Results

About 235 results (0.14 seconds)



Look for the word “backup” in the resulting page, you will be able to locate the command for the backup.



ETCDCTL_API=3 etcdctl — endpoints $ENDPOINT snapshot save snapshotdb

Now wait, this is not the full command that we saw in the beginning. There are some missing parts. Should I memorize the rest? No, run “*ETCDCTL_API=3 etcdctl help*” you will see all the options, you can recognize the missed options here.

```
OPTIONS:
      --cacert=""                          verify certificates of TLS-enabled secure servers using th
is CA bundle
      --cert=""                            identify secure client using this TLS certificate file
      --command-timeout=5s                 timeout for short running command (excluding dial timeout)
      --debug[=false]                      enable client-side debug logging
      --dial-timeout=2s                    dial timeout for client connections
  -d, --discovery-srv=""                   domain name to query for SRV records describing cluster en
dpoints
      --endpoints=[127.0.0.1:2379]         gRPC endpoints
      --hex[=false]                        print byte strings as hex encoded strings
      --insecure-discovery[=true]          accept insecure SRV records describing cluster endpoints
      --insecure-skip-tls-verify[=false]   skip server certificate verification
      --insecure-transport[=true]          disable transport security for client connections
      --keepalive-time=2s                  keepalive time for client connections
      --keepalive-timeout=6s               keepalive timeout for client connections
      --key=""                             identify secure client using this TLS key file
```

Tips 3: Finding the Values

1. Exam cluster setup is done with *kubeadm*, this means ETCD used by the kubernetes cluster is coming from static pod. Confirm this by looking into pods in kube-system namespace.

> kubectl get pod -n kube-system

```
master $
master $ kubectl get pod -n kube-system
NAME                              READY   STATUS    RESTARTS   AGE
coredns-5644d7b6d9-lm986          1/1     Running   0          52m
coredns-5644d7b6d9-nhjbz          1/1     Running   0          52m
etcd-master                       1/1     Running   0          51m
kube-apiserver-master             1/1     Running   0          51m
kube-controller-manager-master    1/1     Running   0          51m
kube-proxy-4bz98                  1/1     Running   0          52m
kube-proxy-556jw                  1/1     Running   0          52m
kube-scheduler-master             1/1     Running   0          51m
weave-net-kkv8v                   2/2     Running   0          52m
weave-net-vcz7z                   2/2     Running   0          52m
```

2. Once you recognize the pod in *kube-system* namespace, just describe the pod to see command line options from *container* section.

> kubectl describe pod etcd-master -n kube-system

```
master $ kubectl describe pod etcd-master -n kube-system
Name:                 etcd-master
Namespace:            kube-system
Priority:             2000000000
Priority Class Name:  system-cluster-critical
Node:                 master/172.17.0.15
Start Time:           Fri, 06 Dec 2019 04:34:46 +0000
Labels:               component=etcd
                      tier=control-plane
Annotations:          kubernetes.io/config.hash: 06b1857c38f07729d4b7304661a28e89
                      kubernetes.io/config.mirror: 06b1857c38f07729d4b7304661a28e89
                      kubernetes.io/config.seen: 2019-12-06T04:34:43.326216315Z
```

```
Containers:
  etcd:
    Container ID:  docker://82ed2eb58ee45e5c00b1200026e163b52ede675ba8f52ebae4c155f0f2804d4a
    Image:         k8s.gcr.io/etcd:3.3.15-0
    Image ID:      docker-pullable://k8s.gcr.io/etcd@sha256:12c2c5e5731c3bcd56e6f1c05c0f9198b6f06793fa7fca2fb43aab9622dc4afa
    Port:          <none>
    Host Port:     <none>
    Command:
      etcd
      --advertise-client-urls=https://172.17.0.15:2379
      --cert-file=/etc/kubernetes/pki/etcd/server.crt
```

```
    --cert-file=/etc/kubernetes/pki/etcd/server.crt
    --client-cert-auth=true
    --data-dir=/var/lib/etcd
    --initial-advertise-peer-urls=https://172.17.0.15:2380
    --initial-cluster=master=https://172.17.0.15:2380
    --key-file=/etc/kubernetes/pki/etcd/server.key
    --listen-client-urls=https://127.0.0.1:2379,https://172.17.0.15:2379
    --listen-metrics-urls=http://127.0.0.1:2381
    --listen-peer-urls=https://172.17.0.15:2380
    --name=master
    --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
    Started:      Fri, 06 Dec 2019 04:34:48 +0000
  Ready:          True
  Restart Count:  0
```

You can locate the information on

1. **endpoint:** — advertise-client-urls=https://172.17.0.15:2379
2. **ca certificate:** — trusted-ca-file=/etc/kubernetes/pki/etcd/ca.crt
3. **server certificate :** — cert-file=/etc/kubernetes/pki/etcd/server.crt
4. **key:** — key-file=/etc/kubernetes/pki/etcd/server.key

Tips 4: Difference in Option Names [IMP]

Please note that the command option name, you get from pod describes and actual "*ETCDCTL_API=3 etcdctl*" are different.

Search this file…

Backup Command	Pod Describe Command
endpoints	advertise-client-urls
cacert	trusted-ca-file
cert	cert-file
key	key-file

test.csv hosted with ♡ by GitHub view raw

You are all done. The ETCD back will be in the specified location.

Also, visit other tips and tricks for *Certified Kubernetes Administrator (CKA)*

Certified Kubernetes Administrator (CKA) — Tips and Tricks