

Advanced Kubernetes

Lab 8 - etcd operations

Etcd is a highly consistent, distributed key-value store designed to reliably and quickly preserve and provide access to critical data. It enables distributed coordination through distributed locking, leader elections, and write barriers. An etcd cluster is intended for high availability and permanent data storage and retrieval.

In this lab we will setup and explore operations on a multi-node etcd cluster. Rather than starting individual computers for each of the etcd nodes, we will run the nodes in docker containers, simulating a three node cluster.

During the lab you will:

- Startup three Ubuntu host containers
- Acquire the latest etcd binaries
- Install etcd in each of the containers
- Start a 3 node cluster
- Test the cluster with etcdctl
- Explore the features of etcdctl
- Add and remove nodes from the cluster
- Backup and restore nodes

1. Starting the etcd cluster containers

To simulate a multi node etcd cluster we will run three docker Ubuntu containers, treating each one like a separate host. We will name the containers:

- nodea
- nodeb
- nodec

Run the three target containers:

```

ubuntu@nodea:~$ docker container run -itd --name nodea ubuntu

Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
6b98dfc16071: Pull complete
4001a1209541: Pull complete
6319fc68c576: Pull complete
b24603670dc3: Pull complete
97f170c87c6f: Pull complete
Digest: sha256:5f4bdc3467537cbbe563e80db2c3ec95d548a9145d64453b06939c4592d67b6d
Status: Downloaded newer image for ubuntu:latest
d502df311675a35fb0df72466d1aef8ac893c572c8d46057d2d0ea73b6941120

ubuntu@nodea:~$ docker container run -itd --name nodeb ubuntu

c2526b9312426661880c21cca8bef389e0eb5c8c17fdb43bde1b1cca8dbf753e

ubuntu@nodea:~$ docker container run -itd --name nodec ubuntu

4300cffff2031c4481abd068b5344d738dee14166f5401bce49d7bf5b3427063a
ubuntu@nodea:~$

```

Verify all three containers are up and running:

```

ubuntu@nodea:~$ docker container ls -f ancestor=ubuntu

CONTAINER ID    IMAGE    COMMAND    CREATED    STATUS    NAMES
4300cffff2031  ubuntu  "/bin/bash"  5 seconds ago  Up 4 seconds  nodec
c2526b931242  ubuntu  "/bin/bash"  12 seconds ago  Up 10 seconds  nodeb
d502df311675  ubuntu  "/bin/bash"  18 seconds ago  Up 17 seconds  nodea
ubuntu@nodea:~$

```

By running the containers with the `-i` and `-t` switches we add support for console input from our host shell and by using the `-d` switch the container is launched "detached" in the background. To connect to the container shells later we can use the `docker container attach` command and to detach from the container shells we can use the `^P ^Q` command sequence.

Next list the IP addresses of each of the containers:

```
ubuntu@nodea:~$ docker container inspect nodea -f '{{.NetworkSettings.IPAddress}}'
172.17.0.2

ubuntu@nodea:~$ docker container inspect nodeb -f '{{.NetworkSettings.IPAddress}}'
172.17.0.3

ubuntu@nodea:~$ docker container inspect nodec -f '{{.NetworkSettings.IPAddress}}'
172.17.0.4
ubuntu@nodea:~$
```

The IP for nodea is .2, nodeb is .3 and nodec is .4 in the example above. We will need these IP addresses to tell our etcd nodes how to find each other.

2. Install etcd in the containers

Installing etcd is as easy as copying it onto the system path, in our case we'll copy it into each of the containers:

```
ubuntu@nodea:~$ docker container cp etcd-v3.3.10-linux-amd64/etcd nodea:/usr/bin/etcd

ubuntu@nodea:~$ docker container exec nodea ls -l /usr/bin/etcd

-rwxr-xr-x 1 1000 1000 17458592 Jun 15 16:56 /usr/bin/etcd
ubuntu@nodea:~$
```

Install etcd on the other two nodes:

```
ubuntu@nodea:~$ docker container cp etcd-v3.3.10-linux-amd64/etcd nodeb:/usr/bin/etcd

ubuntu@nodea:~$ docker container cp etcd-v3.3.10-linux-amd64/etcd nodec:/usr/bin/etcd

ubuntu@nodea:~$
```

Great, etcd is now installed and ready to run on all three of our "computers".

3. Start the first node in the cluster

Obviously we can not start all three nodes at once so we will have to bootstrap our cluster. When launching the nodes in a new cluster we will need to use the following switches:

- `--name nodea` - this defines the human readable node name
- `--initial-advertise-peer-urls http://172.17.0.2:2380` - this defines the address to share with peers
- `--listen-peer-urls http://172.17.0.2:2380` - this defines the address to listen on for peer traffic
- `--listen-client-urls http://172.17.0.2:2379,http://127.0.0.1:2379` - this defines the address to listen on for clients
- `--advertise-client-urls http://172.17.0.2:2379` - this defines the address to advertise to peers for client traffic
- `--initial-cluster-token cluster-1` - this defines the token for the cluster
- `--initial-cluster nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380` - this defines the nodes in the cluster and their peer addresses and ports
- `--initial-cluster-state new` - this lets the cluster acquire quorum progressively as nodes are initially added

In a new terminal, attach to nodea and start the first node of the cluster:

```
ubuntu@nodea:~$ docker container attach nodea
```

```
root@5d3641d008b7:/# etcd --name nodea \
--initial-advertise-peer-urls http://172.17.0.2:2380 \
--listen-peer-urls http://172.17.0.2:2380 \
--listen-client-urls http://172.17.0.2:2379,http://127.0.0.1:2379 \
--advertise-client-urls http://172.17.0.2:2379 \
--initial-cluster-token cluster-1 \
--initial-cluster nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380 \
--initial-cluster-state new
```

```
2019-03-28 01:32:44.764036 I | etcdmain: etcd Version: 3.3.10
2019-03-28 01:32:44.764084 I | etcdmain: Git SHA: 27fc7e2
2019-03-28 01:32:44.764091 I | etcdmain: Go Version: go1.10.4
2019-03-28 01:32:44.764096 I | etcdmain: Go OS/Arch: linux/amd64
2019-03-28 01:32:44.764101 I | etcdmain: setting maximum number of CPUs to 2, total number of available CPUs is 2
2019-03-28 01:32:44.764111 W | etcdmain: no data-dir provided, using default data-dir ./nodea.etcd
2019-03-28 01:32:44.764252 I | embed: listening for peers on http://172.17.0.2:2380
2019-03-28 01:32:44.764327 I | embed: listening for client requests on 127.0.0.1:2379
2019-03-28 01:32:44.764349 I | embed: listening for client requests on 172.17.0.2:2379
2019-03-28 01:32:44.766093 I | etcdserver: name = nodea
2019-03-28 01:32:44.766118 I | etcdserver: data dir = nodea.etcd
```

```

2019-03-28 01:32:44.766123 I | etcdserver: member dir = nodea.etcd/member
2019-03-28 01:32:44.766125 I | etcdserver: heartbeat = 100ms
2019-03-28 01:32:44.766128 I | etcdserver: election = 1000ms
2019-03-28 01:32:44.766130 I | etcdserver: snapshot count = 100000
2019-03-28 01:32:44.766180 I | etcdserver: advertise client URLs = http://172.17.0.2:2379
2019-03-28 01:32:44.766201 I | etcdserver: initial advertise peer URLs = http://172.17.0.2:2380
2019-03-28 01:32:44.766211 I | etcdserver: initial cluster =
nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380
2019-03-28 01:32:44.775574 I | etcdserver: starting member 5c1954e5cd7a3e68 in cluster 89c79295f798999a
2019-03-28 01:32:44.775607 I | raft: 5c1954e5cd7a3e68 became follower at term 0
2019-03-28 01:32:44.775616 I | raft: newRaft 5c1954e5cd7a3e68 [peers: [], term: 0, commit: 0, applied: 0,
lastindex: 0, lastterm: 0]
2019-03-28 01:32:44.775621 I | raft: 5c1954e5cd7a3e68 became follower at term 1
2019-03-28 01:32:44.787555 W | auth: simple token is not cryptographically signed
2019-03-28 01:32:44.788180 I | rafthttp: starting peer 507ecf5e9ca6b606...
2019-03-28 01:32:44.788206 I | rafthttp: started HTTP pipelining with peer 507ecf5e9ca6b606
2019-03-28 01:32:44.789790 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (writer)
2019-03-28 01:32:44.789877 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (writer)
2019-03-28 01:32:44.790242 I | rafthttp: started peer 507ecf5e9ca6b606
2019-03-28 01:32:44.790275 I | rafthttp: added peer 507ecf5e9ca6b606
2019-03-28 01:32:44.790294 I | rafthttp: starting peer 69b68fd2de47b0f9...
2019-03-28 01:32:44.790311 I | rafthttp: started HTTP pipelining with peer 69b68fd2de47b0f9
2019-03-28 01:32:44.790550 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (stream MsgApp v2 reader)
2019-03-28 01:32:44.791081 I | rafthttp: started peer 69b68fd2de47b0f9
2019-03-28 01:32:44.791103 I | rafthttp: added peer 69b68fd2de47b0f9
2019-03-28 01:32:44.791119 I | etcdserver: starting server... [version: 3.3.10, cluster version: to_be_decided]
2019-03-28 01:32:44.791572 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (stream Message reader)
2019-03-28 01:32:44.791589 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 01:32:44.791603 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 01:32:44.791617 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (stream MsgApp v2 reader)
2019-03-28 01:32:44.791760 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (stream Message reader)
2019-03-28 01:32:44.792524 I | etcdserver/membership: added member 507ecf5e9ca6b606 [http://172.17.0.3:2380] to
cluster 89c79295f798999a
2019-03-28 01:32:44.792586 I | etcdserver/membership: added member 5c1954e5cd7a3e68 [http://172.17.0.2:2380] to
cluster 89c79295f798999a
2019-03-28 01:32:44.792621 I | etcdserver/membership: added member 69b68fd2de47b0f9 [http://172.17.0.4:2380] to
cluster 89c79295f798999a
2019-03-28 01:32:45.876608 I | raft: 5c1954e5cd7a3e68 is starting a new election at term 1

```

...

Reading the log output answer these questions:

- What data directory did etcd create?
- What two IP addresses are listening for client connections?
- Why would these two IPs be used for clients?
- How many WAL entries will the server record before taking a snapshot?
- What is nodea's member ID?
- What is your cluster ID?
- What are the IDs of the other two peer nodes?
- Are any of the leader elections that nodea is starting completing?

We will **leave the terminal for nodea open** so that we can monitor its activity. In a new terminal explore the data directory for nodea:

```
ubuntu@nodea:~$ docker container exec nodea ls -l /nodea.etcd

total 4
drwx----- 4 root root 4096 Jul 19 00:20 member

ubuntu@nodea:~$ docker container exec nodea ls -l /nodea.etcd/member

total 8
drwx----- 2 root root 4096 Jul 19 00:20 snap
drwx----- 2 root root 4096 Jul 19 00:20 wal

ubuntu@nodea:~$
```

So at this stage, nodea is started but the cluster is not operational because a leader can not be elected with out a majority of nodes in agreement. Let's start a second node!

4. Start the second node in the cluster

Start a new terminal and launch nodeb with the same switches as nodea, changing the name to nodeb and updating the IP addresses for nodeb:

```
ubuntu@nodea:~$ docker container attach nodeb

root@085a182cc5f7:/# etcd --name nodeb \
```

```

--initial-advertise-peer-urls http://172.17.0.3:2380 \
--listen-peer-urls http://172.17.0.3:2380 \
--listen-client-urls http://172.17.0.3:2379,http://127.0.0.1:2379 \
--advertise-client-urls http://172.17.0.3:2379 \
--initial-cluster-token cluster-1 \
--initial-cluster nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380 \
--initial-cluster-state new

```

```

2019-03-28 01:36:03.053840 I | etcdmain: etcd Version: 3.3.10
2019-03-28 01:36:03.053901 I | etcdmain: Git SHA: 27fc7e2
2019-03-28 01:36:03.053904 I | etcdmain: Go Version: go1.10.4
2019-03-28 01:36:03.053907 I | etcdmain: Go OS/Arch: linux/amd64
2019-03-28 01:36:03.053910 I | etcdmain: setting maximum number of CPUs to 2, total number of available CPUs is 2
2019-03-28 01:36:03.053917 W | etcdmain: no data-dir provided, using default data-dir ./nodeb.etcd
2019-03-28 01:36:03.053993 I | embed: listening for peers on http://172.17.0.3:2380
2019-03-28 01:36:03.054015 I | embed: listening for client requests on 127.0.0.1:2379
2019-03-28 01:36:03.054028 I | embed: listening for client requests on 172.17.0.3:2379
2019-03-28 01:36:03.059658 I | etcdserver: name = nodeb
2019-03-28 01:36:03.059691 I | etcdserver: data dir = nodeb.etcd
2019-03-28 01:36:03.059697 I | etcdserver: member dir = nodeb.etcd/member
2019-03-28 01:36:03.059699 I | etcdserver: heartbeat = 100ms
2019-03-28 01:36:03.059702 I | etcdserver: election = 1000ms
2019-03-28 01:36:03.059704 I | etcdserver: snapshot count = 100000
2019-03-28 01:36:03.059711 I | etcdserver: advertise client URLs = http://172.17.0.3:2379
2019-03-28 01:36:03.059716 I | etcdserver: initial advertise peer URLs = http://172.17.0.3:2380
2019-03-28 01:36:03.059724 I | etcdserver: initial cluster =
nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380
2019-03-28 01:36:03.062813 I | etcdserver: starting member 507ecf5e9ca6b606 in cluster 89c79295f798999a
2019-03-28 01:36:03.062865 I | raft: 507ecf5e9ca6b606 became follower at term 0
2019-03-28 01:36:03.062876 I | raft: newRaft 507ecf5e9ca6b606 [peers: [], term: 0, commit: 0, applied: 0,
lastindex: 0, lastterm: 0]
2019-03-28 01:36:03.062880 I | raft: 507ecf5e9ca6b606 became follower at term 1
2019-03-28 01:36:03.065253 W | auth: simple token is not cryptographically signed
2019-03-28 01:36:03.066048 I | rafthttp: starting peer 5c1954e5cd7a3e68...
2019-03-28 01:36:03.066091 I | rafthttp: started HTTP pipelining with peer 5c1954e5cd7a3e68
2019-03-28 01:36:03.066506 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (writer)
2019-03-28 01:36:03.066780 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (writer)
2019-03-28 01:36:03.067621 I | rafthttp: started peer 5c1954e5cd7a3e68
2019-03-28 01:36:03.067749 I | rafthttp: added peer 5c1954e5cd7a3e68
2019-03-28 01:36:03.067785 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (stream MsgApp v2 reader)
2019-03-28 01:36:03.067874 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (stream Message reader)
2019-03-28 01:36:03.068026 I | rafthttp: starting peer 69b68fd2de47b0f9...
2019-03-28 01:36:03.068072 I | rafthttp: started HTTP pipelining with peer 69b68fd2de47b0f9

```

```

2019-03-28 01:36:03.069607 I | rafthttp: peer 5c1954e5cd7a3e68 became active
2019-03-28 01:36:03.069622 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
Message reader)
2019-03-28 01:36:03.069681 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
MsgApp v2 reader)
2019-03-28 01:36:03.073821 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 01:36:03.074356 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 01:36:03.074657 I | rafthttp: started peer 69b68fd2de47b0f9
2019-03-28 01:36:03.074759 I | rafthttp: added peer 69b68fd2de47b0f9
2019-03-28 01:36:03.074847 I | etcdserver: starting server... [version: 3.3.10, cluster version: to_be_decided]
2019-03-28 01:36:03.074942 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (stream MsgApp v2 reader)
2019-03-28 01:36:03.075221 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (stream Message reader)
2019-03-28 01:36:03.076402 I | etcdserver/membership: added member 507ecf5e9ca6b606 [http://172.17.0.3:2380] to
cluster 89c79295f798999a
2019-03-28 01:36:03.076616 I | etcdserver/membership: added member 5c1954e5cd7a3e68 [http://172.17.0.2:2380] to
cluster 89c79295f798999a
2019-03-28 01:36:03.076749 I | etcdserver/membership: added member 69b68fd2de47b0f9 [http://172.17.0.4:2380] to
cluster 89c79295f798999a
2019-03-28 01:36:03.093743 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
MsgApp v2 writer)
2019-03-28 01:36:03.099027 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
Message writer)
2019-03-28 01:36:03.678824 I | raft: 507ecf5e9ca6b606 [term: 1] received a MsgVote message with higher term from
5c1954e5cd7a3e68 [term: 138]
2019-03-28 01:36:03.678887 I | raft: 507ecf5e9ca6b606 became follower at term 138
2019-03-28 01:36:03.678902 I | raft: 507ecf5e9ca6b606 [logterm: 1, index: 3, vote: 0] cast MsgVote for
5c1954e5cd7a3e68 [logterm: 1, index: 3] at term 138
2019-03-28 01:36:03.680701 I | raft: raft.node: 507ecf5e9ca6b606 elected leader 5c1954e5cd7a3e68 at term 138
2019-03-28 01:36:03.682985 I | etcdserver: published {Name:nodeb ClientURLs:[http://172.17.0.3:2379]} to cluster
89c79295f798999a
2019-03-28 01:36:03.683134 I | embed: ready to serve client requests
2019-03-28 01:36:03.684080 N | embed: serving insecure client requests on 127.0.0.1:2379, this is strongly
discouraged!
2019-03-28 01:36:03.684129 I | embed: ready to serve client requests
2019-03-28 01:36:03.684788 N | embed: serving insecure client requests on 172.17.0.3:2379, this is strongly
discouraged!
2019-03-28 01:36:03.684917 N | etcdserver/membership: set the initial cluster version to 3.0
2019-03-28 01:36:03.684969 I | etcdserver/api: enabled capabilities for version 3.0

```

...

Reading the nodeb log answer the following questions:

- Does the nodeb ID reported by nodeb match the one of the IDs reported by nodea?
- Locate the line with this text: "peer xxxxxxxxxxxx became active", where xxxxxxxxxxxx is some member ID. Which node is this line referring to? What is it telling us?
- Locate the line with this text "received a MsgVote message with higher term from". Which member is referred to in this line? Is nodeb the leader?
- Is nodeb ready to serve clients?
- Examine the new log output on nodea, what was the initial cluster version set to?

Great, we now have an active cluster with 2 nodes! We are in the danger zone however, because if we lose a node we lose quorum and our cluster goes down. Let's add the third node.

5. Start the final node in the cluster

In another new terminal launch nodec with the same switches as nodea and nodeb, changing the name to nodec and updating the IP addresses for nodec:

```
ubuntu@nodea:~$ docker container attach nodec
```

```
root@66488ce9febb:/# etcd --name nodec \  
--initial-advertise-peer-urls http://172.17.0.4:2380 \  
--listen-peer-urls http://172.17.0.4:2380 \  
--listen-client-urls http://172.17.0.4:2379,http://127.0.0.1:2379 \  
--advertise-client-urls http://172.17.0.4:2379 \  
--initial-cluster-token cluster-1 \  
--initial-cluster nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380 \  
--initial-cluster-state new
```

```
2019-03-28 01:38:43.121149 I | etcdmain: etcd Version: 3.3.10  
2019-03-28 01:38:43.121229 I | etcdmain: Git SHA: 27fc7e2  
2019-03-28 01:38:43.121234 I | etcdmain: Go Version: go1.10.4  
2019-03-28 01:38:43.121236 I | etcdmain: Go OS/Arch: linux/amd64  
2019-03-28 01:38:43.121239 I | etcdmain: setting maximum number of CPUs to 2, total number of available CPUs is 2  
2019-03-28 01:38:43.121244 W | etcdmain: no data-dir provided, using default data-dir ./nodec.etcd  
2019-03-28 01:38:43.121299 I | embed: listening for peers on http://172.17.0.4:2380  
2019-03-28 01:38:43.121320 I | embed: listening for client requests on 127.0.0.1:2379  
2019-03-28 01:38:43.121331 I | embed: listening for client requests on 172.17.0.4:2379  
2019-03-28 01:38:43.124493 I | etcdserver: name = nodec  
2019-03-28 01:38:43.124517 I | etcdserver: data dir = nodec.etcd  
2019-03-28 01:38:43.124521 I | etcdserver: member dir = nodec.etcd/member  
2019-03-28 01:38:43.124524 I | etcdserver: heartbeat = 100ms
```

```

2019-03-28 01:38:43.124526 I | etcdserver: election = 1000ms
2019-03-28 01:38:43.124532 I | etcdserver: snapshot count = 100000
2019-03-28 01:38:43.124538 I | etcdserver: advertise client URLs = http://172.17.0.4:2379
2019-03-28 01:38:43.124541 I | etcdserver: initial advertise peer URLs = http://172.17.0.4:2380
2019-03-28 01:38:43.124551 I | etcdserver: initial cluster =
nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,nodec=http://172.17.0.4:2380
2019-03-28 01:38:43.127459 I | etcdserver: starting member 69b68fd2de47b0f9 in cluster 89c79295f798999a
2019-03-28 01:38:43.127509 I | raft: 69b68fd2de47b0f9 became follower at term 0
2019-03-28 01:38:43.127517 I | raft: newRaft 69b68fd2de47b0f9 [peers: [], term: 0, commit: 0, applied: 0,
lastindex: 0, lastterm: 0]
2019-03-28 01:38:43.127521 I | raft: 69b68fd2de47b0f9 became follower at term 1
2019-03-28 01:38:43.128872 W | auth: simple token is not cryptographically signed
2019-03-28 01:38:43.129312 I | rafthttp: starting peer 507ecf5e9ca6b606...
2019-03-28 01:38:43.129334 I | rafthttp: started HTTP pipelining with peer 507ecf5e9ca6b606
2019-03-28 01:38:43.129787 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (writer)
2019-03-28 01:38:43.130271 I | rafthttp: started peer 507ecf5e9ca6b606
2019-03-28 01:38:43.130373 I | rafthttp: added peer 507ecf5e9ca6b606
2019-03-28 01:38:43.130470 I | rafthttp: starting peer 5c1954e5cd7a3e68...
2019-03-28 01:38:43.130561 I | rafthttp: started HTTP pipelining with peer 5c1954e5cd7a3e68
2019-03-28 01:38:43.132909 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (writer)
2019-03-28 01:38:43.133347 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (stream MsgApp v2 reader)
2019-03-28 01:38:43.134211 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (stream Message reader)
2019-03-28 01:38:43.137610 I | rafthttp: peer 507ecf5e9ca6b606 became active
2019-03-28 01:38:43.137755 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
MsgApp v2 reader)
2019-03-28 01:38:43.138582 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
Message reader)
2019-03-28 01:38:43.140371 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (writer)
2019-03-28 01:38:43.147437 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (writer)
2019-03-28 01:38:43.148463 I | rafthttp: started peer 5c1954e5cd7a3e68
2019-03-28 01:38:43.149762 I | rafthttp: added peer 5c1954e5cd7a3e68
2019-03-28 01:38:43.150067 I | etcdserver: starting server... [version: 3.3.10, cluster version: to_be_decided]
2019-03-28 01:38:43.151298 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (stream Message reader)
2019-03-28 01:38:43.152114 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (stream MsgApp v2 reader)
2019-03-28 01:38:43.152666 I | etcdserver/membership: added member 507ecf5e9ca6b606 [http://172.17.0.3:2380] to
cluster 89c79295f798999a
2019-03-28 01:38:43.153004 I | etcdserver/membership: added member 5c1954e5cd7a3e68 [http://172.17.0.2:2380] to
cluster 89c79295f798999a
2019-03-28 01:38:43.153183 I | etcdserver/membership: added member 69b68fd2de47b0f9 [http://172.17.0.4:2380] to
cluster 89c79295f798999a
2019-03-28 01:38:43.153869 I | rafthttp: peer 5c1954e5cd7a3e68 became active
2019-03-28 01:38:43.153887 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
Message reader)

```

```

2019-03-28 01:38:43.154044 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
MsgApp v2 reader)
2019-03-28 01:38:43.176315 I | raft: 69b68fd2de47b0f9 [term: 1] received a MsgHeartbeat message with higher term
from 5c1954e5cd7a3e68 [term: 138]
2019-03-28 01:38:43.176354 I | raft: 69b68fd2de47b0f9 became follower at term 138
2019-03-28 01:38:43.176364 I | raft: raft.node: 69b68fd2de47b0f9 elected leader 5c1954e5cd7a3e68 at term 138
2019-03-28 01:38:43.176698 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
Message writer)
2019-03-28 01:38:43.176816 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
MsgApp v2 writer)
2019-03-28 01:38:43.178080 N | etcdserver/membership: set the initial cluster version to 3.0
2019-03-28 01:38:43.178233 I | etcdserver/api: enabled capabilities for version 3.0
2019-03-28 01:38:43.178640 I | etcdserver: published {Name:nodec ClientURLs:[http://172.17.0.4:2379]} to cluster
89c79295f798999a
2019-03-28 01:38:43.178873 I | embed: ready to serve client requests
2019-03-28 01:38:43.179012 I | embed: ready to serve client requests
2019-03-28 01:38:43.179417 N | embed: serving insecure client requests on 172.17.0.4:2379, this is strongly
discouraged!
2019-03-28 01:38:43.179586 N | embed: serving insecure client requests on 127.0.0.1:2379, this is strongly
discouraged!
2019-03-28 01:38:43.193392 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
Message writer)
2019-03-28 01:38:43.194787 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
MsgApp v2 writer)
2019-03-28 01:38:43.205360 I | etcdserver: 69b68fd2de47b0f9 initialized peer connection; fast-forwarding 8 ticks
(election ticks 10) with 2 active peer(s)
2019-03-28 01:38:43.747895 N | etcdserver/membership: updated the cluster version from 3.0 to 3.3
2019-03-28 01:38:43.748056 I | etcdserver/api: enabled capabilities for version 3.3

```

Reading the log output from nodec answer the following questions:

- What is the ID of nodec
- Did nodec vote to elect nodea leader?
- How many of the other nodes is nodec connected to?
- What was the cluster version updated to? Why do you think this happened?

Fantastic, we have a fully operational cluster running!

6. Connecting to the cluster

Return to a host terminal (or create a new one). We will use the `etcdctl` tool to exercise our new cluster. To make operations easier let's copy the `etcdctl` binary to the system path:

```
ubuntu@nodea:~$ sudo cp etcd/etcdctl /usr/bin
ubuntu@nodea:~$
```

Our cluster is running `etcd v3` so we will want to set the `v3` environment variable so that `etcdctl` uses the right protocol. Set the environment variable and test `etcdctl`:

```
ubuntu@nodea:~$ export ETCDCTL_API=3
ubuntu@nodea:~$ etcdctl version
etcdctl version: 3.3.10
API version: 3.3
ubuntu@nodea:~$
```

In order to access the cluster we will need to provide `etcdctl` with at least one of the cluster client endpoints. However if we give `etcdctl` all of the client endpoints, even if a node is down we will still be able to interact with the remaining nodes.

Try listing the cluster members using the client endpoint list for the entire cluster:

```
ubuntu@nodea:~$ etcdctl --endpoints=[172.17.0.2:2379,172.17.0.3:2379,172.17.0.4:2379] member list
507ecf5e9ca6b606, started, nodeb, http://172.17.0.3:2380, http://172.17.0.3:2379
5c1954e5cd7a3e68, started, nodea, http://172.17.0.2:2380, http://172.17.0.2:2379
69b68fd2de47b0f9, started, nodec, http://172.17.0.4:2380, http://172.17.0.4:2379
ubuntu@nodea:~$
```

It works! Now display the help for the member list command:

```
ubuntu@nodea:~$ etcdctl --endpoints=[172.17.0.2:2379,172.17.0.3:2379,172.17.0.4:2379] member list --help
```

NAME:

member list - Lists all members in the cluster

USAGE:

etcdctl member list

DESCRIPTION:

When `--write-out` is set to `simple`, this command prints out comma-separated member lists for each endpoint. The items in the lists are ID, Status, Name, Peer Addr, Client Addr.

GLOBAL OPTIONS:

<code>--cacert=""</code>	verify certificates of TLS-enabled secure servers using this CA bundle
<code>--cert=""</code>	identify secure client using this TLS certificate file
<code>--command-timeout=5s</code>	timeout for short running command (excluding dial timeout)
<code>--debug[=false]</code>	enable client-side debug logging
<code>--dial-timeout=2s</code>	dial timeout for client connections
<code>-d, --discovery-srv=""</code>	domain name to query for SRV records describing cluster endpoints
<code>--endpoints=[127.0.0.1:2379]</code>	gRPC endpoints
<code>--hex[=false]</code>	print byte strings as hex encoded strings
<code>--insecure-discovery[=true]</code>	accept insecure SRV records describing cluster endpoints
<code>--insecure-skip-tls-verify[=false]</code>	skip server certificate verification
<code>--insecure-transport[=true]</code>	disable transport security for client connections
<code>--keepalive-time=2s</code>	keepalive time for client connections
<code>--keepalive-timeout=6s</code>	keepalive timeout for client connections
<code>--key=""</code>	identify secure client using this TLS key file
<code>--user=""</code>	username[:password] for authentication (prompt if password is not supplied)
<code>-w, --write-out="simple"</code>	set the output format (fields, json, protobuf, simple, table)

```
ubuntu@nodea:~$
```

The member list command supports JSON output. Try it:

```
ubuntu@nodea:~$ etcdctl --endpoints=[172.17.0.2:2379,172.17.0.3:2379,172.17.0.4:2379] member list -w="json"

{"header":{"cluster_id":"9928065076363303322","member_id":"5800301375361824262","raft_term":208},"members":
[{"ID":"5800301375361824262","name":"nodeb","peerURLs":["http://172.17.0.3:2380"],"clientURLs":
["http://172.17.0.3:2379"]}, {"ID":"6636428871878721128","name":"nodea","peerURLs":
```

```
["http://172.17.0.2:2380"], "clientURLs": ["http://172.17.0.2:2379"]},  
{ "ID": "7617433955578917113", "name": "nodec", "peerURLs": ["http://172.17.0.4:2380"], "clientURLs":  
["http://172.17.0.4:2379"]}]}
```

ubuntu@nodea:~\$

Nice, now we can extract machine readable data from etcd. However, it is not fun entering in the end point addresses with each command. We can set an environment variable for the cluster end points so simplify things. Try it:

```
ubuntu@nodea:~$ export ETCDCTL_ENDPOINTS=172.17.0.2:2379,172.17.0.3:2379,172.17.0.4:2379  
  
ubuntu@nodea:~$ etcdctl member list  
  
507ecf5e9ca6b606, started, nodeb, http://172.17.0.3:2380, http://172.17.0.3:2379  
5c1954e5cd7a3e68, started, nodea, http://172.17.0.2:2380, http://172.17.0.2:2379  
69b68fd2de47b0f9, started, nodec, http://172.17.0.4:2380, http://172.17.0.4:2379  
ubuntu@nodea:~$
```

Much better. Now run a health check on the cluster:

```
ubuntu@nodea:~$ etcdctl endpoint health  
  
172.17.0.3:2379 is healthy: successfully committed proposal: took = 5.52107ms  
172.17.0.2:2379 is healthy: successfully committed proposal: took = 12.425642ms  
172.17.0.4:2379 is healthy: successfully committed proposal: took = 18.139779ms  
ubuntu@nodea:~$
```

Everything looks good. Let's try saving a key/value pair:

```
ubuntu@nodea:~$ etcdctl put testport 7777  
  
OK  
  
ubuntu@nodea:~$ etcdctl get testport  
  
testport  
7777
```

```
ubuntu@nodea:~$
```

Our etcd cluster is working!

Examine the help and try some other get command variants:

```
ubuntu@nodea:~$ etcdctl get --help
```

NAME:

get - Gets the key or a range of keys

USAGE:

etcdctl get [options] <key> [range_end]

OPTIONS:

--consistency=""	Linearizable(l) or Serializable(s)
--from-key[=false]	Get keys that are greater than or equal to the given key using byte
compare	
--keys-only[=false]	Get only the keys
--limit=0	Maximum number of results
--order=""	Order of results; ASCEND or DESCEND (ASCEND by default)
--prefix[=false]	Get keys with matching prefix
--print-value-only[=false]	Only write values when using the "simple" output format
--rev=0	Specify the kv revision
--sort-by=""	Sort target; CREATE, KEY, MODIFY, VALUE, or VERSION

GLOBAL OPTIONS:

...

```
ubuntu@nodea:~$ etcdctl get testport --print-value-only
```

```
7777
```

```
ubuntu@nodea:~$ etcdctl get test --print-value-only --prefix
```

```
7777
```

```
ubuntu@nodea:~$
```

Perfect we have a working cluster and a configured client!

CHALLENGES (optional)

Using the class slides and lecture information as a guide, complete the following tasks:

Remove a node

Remove nodec from the cluster (but leave the container running):

- `etcdctl member list`
- `etcdctl member remove <member_id>`
- Review the logs:
 - `2019-03-28 01:49:35.631421 E | etcdserver: the member has been permanently removed from the cluster`

Test the cluster, does it still work?

- `etcdctl get testport`
- `etcdctl put newvalue 11111`
- `etcdctl get newvalue`

Create a new node

Add noded to the cluster, **in a new terminal**:

- `docker container run -itd --name noded ubuntu`
- `docker container inspect noded -f '{{.NetworkSettings.IPAddress}}'`
- `docker container cp etcd/etcd noded:/usr/bin/etcd`

Back in the **host terminal**, add the new member:

```
etcdctl member add noded --peer-urls=http://172.17.0.5:2380
```

```
Member 44269ebfb910cd5f added to cluster 89c79295f798999a
```

```
ETCD_NAME="noded"  
ETCD_INITIAL_CLUSTER="noded=http://172.17.0.5:2380,nodeb=http://172.17.0.3:2380,nodea=http://172.17.0.2:2380"  
ETCD_INITIAL_ADVERTISE_PEER_URLS="http://172.17.0.5:2380"  
ETCD_INITIAL_CLUSTER_STATE="existing"
```


In the noded terminal, start the node:

```
ubuntu@nodea:~$ docker container attach noded

root@802f0163d27d:/# etcd --name noded \
--initial-advertise-peer-urls http://172.17.0.5:2380 \
--listen-peer-urls http://172.17.0.5:2380 \
--listen-client-urls http://172.17.0.5:2379,http://127.0.0.1:2379 \
--advertise-client-urls http://172.17.0.5:2379 \
--initial-cluster-token cluster-1 \
--initial-cluster nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,noded=http://172.17.0.5:2380 \
--initial-cluster-state existing
```

In the host terminal verify your changes succeeded:

```
ubuntu@nodea:~$ etcdctl member list

44269ebfb910cd5f, started, noded, http://172.17.0.5:2380, http://172.17.0.5:2379
507ecf5e9ca6b606, started, nodeb, http://172.17.0.3:2380, http://172.17.0.3:2379
5c1954e5cd7a3e68, started, nodea, http://172.17.0.2:2380, http://172.17.0.2:2379
ubuntu@nodea:~$
```

Migrate node

Migrate noded to nodec and rejoin it as "noded" with the cluster

Kill the noded etcd process () but leave the container running

In the nodec container remove the old nodec data and make a directory for noded:

```
root@c428a01f4fe9:/# rm -rf nodec.etcd/

root@c428a01f4fe9:/# mkdir noded.etcd/
```

Copy the noded member directory:

```
ubuntu@nodea:~$ docker container cp noded:noded.etcd/member/ .

ubuntu@nodea:~$ ls -l member/
total 16
drwx----- 4 user user 4096 Mar 27 20:32 ./
drwxr-xr-x 19 user user 4096 Mar 27 21:24 ../
drwx----- 2 user user 4096 Mar 27 20:32 snap/
drwx----- 2 user user 4096 Mar 27 20:43 wal/

ubuntu@nodea:~$ docker container cp member/ nodec:/noded.etcd/member/

ubuntu@nodea:~$
```

Confirm it was successful in the nodec container:

```
root@c428a01f4fe9:/# ls -l noded.etcd/member/

total 8
drwx----- 2 1000 1000 4096 Mar 28 03:32 snap
drwx----- 2 1000 1000 4096 Mar 28 03:43 wal
root@c428a01f4fe9:/#
```

Update the member information for noded for the new IP:

```
ubuntu@nodea:~$ etcdctl member list

44269ebfb910cd5f, started, noded, http://172.17.0.5:2380, http://172.17.0.5:2379
507ecf5e9ca6b606, started, nodeb, http://172.17.0.3:2380, http://172.17.0.3:2379
5c1954e5cd7a3e68, started, nodea, http://172.17.0.2:2380, http://172.17.0.2:2379

ubuntu@nodea:~$ etcdctl member update 44269ebfb910cd5f --peer-urls=http://172.17.0.4:2380

Member 44269ebfb910cd5f updated in cluster 89c79295f798999a
ubuntu@nodea:~$
```

Start the "new" noded in the nodec container:

```

root@c428a01f4fe9:/# etcd --name noded \
--initial-advertise-peer-urls http://172.17.0.4:2380 \
--listen-peer-urls http://172.17.0.4:2380 \
--listen-client-urls http://172.17.0.4:2379,http://127.0.0.1:2379 \
--advertise-client-urls http://172.17.0.4:2379 \
--initial-cluster-token cluster-1 \
--initial-cluster nodea=http://172.17.0.2:2380,nodeb=http://172.17.0.3:2380,noded=http://172.17.0.4:2380 \
--initial-cluster-state existing

2019-03-28 04:28:09.782895 I | etcdmain: etcd Version: 3.3.10
2019-03-28 04:28:09.783012 I | etcdmain: Git SHA: 27fc7e2
2019-03-28 04:28:09.783256 I | etcdmain: Go Version: go1.10.4
2019-03-28 04:28:09.783368 I | etcdmain: Go OS/Arch: linux/amd64
2019-03-28 04:28:09.784068 I | etcdmain: setting maximum number of CPUs to 2, total number of available CPUs is 2
2019-03-28 04:28:09.784112 W | etcdmain: no data-dir provided, using default data-dir ./noded.etcd
2019-03-28 04:28:09.784388 N | etcdmain: the server is already initialized as member before, starting as etcd
member...
2019-03-28 04:28:09.784556 I | embed: listening for peers on http://172.17.0.4:2380
2019-03-28 04:28:09.784715 I | embed: listening for client requests on 127.0.0.1:2379
2019-03-28 04:28:09.784886 I | embed: listening for client requests on 172.17.0.4:2379
2019-03-28 04:28:09.786500 I | etcdserver: name = noded
2019-03-28 04:28:09.786592 I | etcdserver: data dir = noded.etcd
2019-03-28 04:28:09.786621 I | etcdserver: member dir = noded.etcd/member
2019-03-28 04:28:09.786654 I | etcdserver: heartbeat = 100ms
2019-03-28 04:28:09.786679 I | etcdserver: election = 1000ms
2019-03-28 04:28:09.786702 I | etcdserver: snapshot count = 100000
2019-03-28 04:28:09.786741 I | etcdserver: advertise client URLs = http://172.17.0.4:2379
2019-03-28 04:28:09.806374 I | etcdserver: restarting member 63d0b0127f21275f in cluster 89c79295f798999a at
commit index 17
2019-03-28 04:28:09.806661 I | raft: 63d0b0127f21275f became follower at term 138
2019-03-28 04:28:09.806769 I | raft: newRaft 63d0b0127f21275f [peers: [], term: 138, commit: 17, applied: 0,
lastindex: 17, lastterm: 138]
2019-03-28 04:28:09.809069 W | auth: simple token is not cryptographically signed
2019-03-28 04:28:09.809547 I | etcdserver: starting server... [version: 3.3.10, cluster version: to_be_decided]
2019-03-28 04:28:09.810759 I | etcdserver/membership: added member 507ecf5e9ca6b606 [http://172.17.0.3:2380] to
cluster 89c79295f798999a
2019-03-28 04:28:09.810793 I | rafthttp: starting peer 507ecf5e9ca6b606...
2019-03-28 04:28:09.810810 I | rafthttp: started HTTP pipelining with peer 507ecf5e9ca6b606
2019-03-28 04:28:09.814323 I | rafthttp: started peer 507ecf5e9ca6b606
2019-03-28 04:28:09.814353 I | rafthttp: added peer 507ecf5e9ca6b606
2019-03-28 04:28:09.814522 I | etcdserver/membership: added member 5c1954e5cd7a3e68 [http://172.17.0.2:2380] to
cluster 89c79295f798999a

```

```

2019-03-28 04:28:09.814535 I | rafthttp: starting peer 5c1954e5cd7a3e68...
2019-03-28 04:28:09.814544 I | rafthttp: started HTTP pipelining with peer 5c1954e5cd7a3e68
2019-03-28 04:28:09.815388 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (writer)
2019-03-28 04:28:09.817619 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (writer)
2019-03-28 04:28:09.817655 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (writer)
2019-03-28 04:28:09.817853 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (stream MsgApp v2 reader)
2019-03-28 04:28:09.818201 I | rafthttp: started streaming with peer 507ecf5e9ca6b606 (stream Message reader)
2019-03-28 04:28:09.825782 I | rafthttp: started peer 5c1954e5cd7a3e68
2019-03-28 04:28:09.825815 I | rafthttp: added peer 5c1954e5cd7a3e68
2019-03-28 04:28:09.827246 I | rafthttp: peer 507ecf5e9ca6b606 became active
2019-03-28 04:28:09.828425 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
Message writer)
2019-03-28 04:28:09.828457 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (writer)
2019-03-28 04:28:09.828464 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (stream MsgApp v2 reader)
2019-03-28 04:28:09.829342 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
MsgApp v2 writer)
2019-03-28 04:28:09.829579 I | rafthttp: peer 5c1954e5cd7a3e68 became active
2019-03-28 04:28:09.829590 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
Message writer)
2019-03-28 04:28:09.830973 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
Message reader)
2019-03-28 04:28:09.831063 I | rafthttp: started streaming with peer 5c1954e5cd7a3e68 (stream Message reader)
2019-03-28 04:28:09.831285 I | etcdserver/membership: added member 69b68fd2de47b0f9 [http://172.17.0.4:2380] to
cluster 89c79295f798999a
2019-03-28 04:28:09.831407 I | rafthttp: starting peer 69b68fd2de47b0f9...
2019-03-28 04:28:09.831526 I | rafthttp: started HTTP pipelining with peer 69b68fd2de47b0f9
2019-03-28 04:28:09.832555 I | rafthttp: established a TCP streaming connection with peer 507ecf5e9ca6b606 (stream
MsgApp v2 reader)
2019-03-28 04:28:09.834832 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
MsgApp v2 reader)
2019-03-28 04:28:09.835220 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
Message reader)
2019-03-28 04:28:09.835369 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 04:28:09.835642 I | rafthttp: started peer 69b68fd2de47b0f9
2019-03-28 04:28:09.835699 I | rafthttp: added peer 69b68fd2de47b0f9
2019-03-28 04:28:09.835750 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 04:28:09.835830 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (stream MsgApp v2 reader)
2019-03-28 04:28:09.836121 I | rafthttp: started streaming with peer 69b68fd2de47b0f9 (stream Message reader)
2019-03-28 04:28:09.836193 N | etcdserver/membership: set the initial cluster version to 3.0
2019-03-28 04:28:09.836272 I | etcdserver/api: enabled capabilities for version 3.0
2019-03-28 04:28:09.836391 N | etcdserver/membership: updated the cluster version from 3.0 to 3.3
2019-03-28 04:28:09.836500 I | etcdserver/api: enabled capabilities for version 3.3
2019-03-28 04:28:09.837128 I | rafthttp: started HTTP pipelining with peer 63d0b0127f21275f

```

```

2019-03-28 04:28:09.837143 E | rafthttp: failed to find member 63d0b0127f21275f in cluster 89c79295f798999a
2019-03-28 04:28:09.837611 E | rafthttp: failed to find member 63d0b0127f21275f in cluster 89c79295f798999a
2019-03-28 04:28:09.839253 I | rafthttp: established a TCP streaming connection with peer 5c1954e5cd7a3e68 (stream
MsgApp v2 writer)
2019-03-28 04:28:09.839272 E | rafthttp: failed to find member 63d0b0127f21275f in cluster 89c79295f798999a
2019-03-28 04:28:09.839639 I | etcdserver/membership: removed member 69b68fd2de47b0f9 from cluster
89c79295f798999a
2019-03-28 04:28:09.839652 I | rafthttp: stopping peer 69b68fd2de47b0f9...
2019-03-28 04:28:09.839659 I | rafthttp: stopped streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 04:28:09.839667 I | rafthttp: stopped streaming with peer 69b68fd2de47b0f9 (writer)
2019-03-28 04:28:09.839680 I | rafthttp: stopped HTTP pipelining with peer 69b68fd2de47b0f9
2019-03-28 04:28:09.839688 I | rafthttp: stopped streaming with peer 69b68fd2de47b0f9 (stream MsgApp v2 reader)
2019-03-28 04:28:09.839693 I | rafthttp: stopped streaming with peer 69b68fd2de47b0f9 (stream Message reader)
2019-03-28 04:28:09.839712 I | rafthttp: stopped peer 69b68fd2de47b0f9
2019-03-28 04:28:09.839718 I | rafthttp: removed peer 69b68fd2de47b0f9
2019-03-28 04:28:09.840016 I | etcdserver/membership: added member 63d0b0127f21275f [http://172.17.0.5:2380] to
cluster 89c79295f798999a
2019-03-28 04:28:09.842379 I | raft: raft.node: 63d0b0127f21275f elected leader 5c1954e5cd7a3e68 at term 138
2019-03-28 04:28:09.873522 I | etcdserver: 63d0b0127f21275f initialized peer connection; fast-forwarding 8 ticks
(election ticks 10) with 2 active peer(s)
2019-03-28 04:28:09.875259 N | etcdserver/membership: updated member 63d0b0127f21275f [http://172.17.0.4:2380] in
cluster 89c79295f798999a
2019-03-28 04:28:09.875787 I | embed: ready to serve client requests
2019-03-28 04:28:09.875886 I | embed: ready to serve client requests
2019-03-28 04:28:09.876274 N | embed: serving insecure client requests on 172.17.0.4:2379, this is strongly
discouraged!
2019-03-28 04:28:09.876334 I | etcdserver: published {Name:noded ClientURLs:[http://172.17.0.4:2379]} to cluster
89c79295f798999a
2019-03-28 04:28:09.876505 N | embed: serving insecure client requests on 127.0.0.1:2379, this is strongly
discouraged!

```

Confirm the new noded is available:

```
ubuntu@nodea:~$ etcdctl member list
```

```

44269ebfb910cd5f, started, noded, http://172.17.0.4:2380, http://172.17.0.4:2379
507ecf5e9ca6b606, started, nodeb, http://172.17.0.3:2380, http://172.17.0.3:2379
5c1954e5cd7a3e68, started, nodea, http://172.17.0.2:2380, http://172.17.0.2:2379

```

```
ubuntu@nodea:~$ etcdctl get newvalue
```

```
newvalue  
11111  
ubuntu@nodea:~$
```

Success!

Congratulations you have completed the lab!

Copyright (c) 2014-2019 RX-M LLC, Cloud Native Consulting, all rights reserved