Etcd based master contender/detector (MESOS-1806)

NOTE: This is a public document

Introduction

Mesos supports running multiple masters at the same time for high availability. To set it up, a zookeeper quorum is required. MESOS-4610 (already merged) makes it possible to load custom contender/detector from modules. MESOS-1806 aims to build etcd contender/detectors module.

Goals

Describe the details of using etcd as backend of master election and detection.

Motivation

With the increasing number of etcd users, for them to use Mesos it's often required to run another zookeeper cluster just for Mesos. It would be ideal for them if mesos can use etcd as an alternative of zookeeper.

The “CompareAndSwap” Operation of Etcd API

Etcd key-value api supports the “CompareAndSwap” semantics, which can be used for distributed leader election.

There are three request parameters the can be used to “compare”:

1. prevExist: checks the previous value of the key.
2. prevIndex: checks the previous modifiedIndex of the key.
3. prevValue: checks the previous value of the key.

More details can be found in [etcd api documentation].

Solution Description

Master Contending and Detecting

The etcd servers uri and the path of the key would be specified in the “--module” flags, with the format like “etcd://host1:port1,host2:port2,host3:port3/v2/keys/mesos”.

When the master starts up, it would first try to read the key from etcd key-value api.

- If the key doesn't exist, it would try to create it. The value is the json-serialized MasterInfo PB.
- If the key exists, it would set a watch on this key, and tries to contend again when the watch is triggered.
- The key is created with a TTL, and the leading master would update the key periodically before the TTL expires. If the leading master dies, the key would be deleted when the TTL expires, thus the other masters would contend again.

The detector (for master/slave/frameworks) works by reading the key, and parse the MasterInfo json object. The detector would then set a watch on the key, and re-read the Masterinfo when the watch is triggered.

Add a new --masters flag to list all the masters

Currently, zookeeper is used for the list of masters in the replicated logs group. When using non-zookeeper contenders, the list of masters must be specified explicitly with “--masters” flags, e.g:

```
mesos-master--masters=10.1.1.1:5050,10.1.1.2:5050,10.1.1.3:5050 ...
```

(can we use a etcd directory for the replication logs group?)