KIP-273 - Kafka to support using ETCD beside Zookeeper

Status

Current state: *Under discussion*

Discussion thread: *here*

JIRA: *KAFKA-6598*

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Motivation

The current Kafka implementation is bound to Zookeeper to store its metadata for forming a cluster of nodes (producer/consumer/broker). As Kafka is becoming popular for streaming in various environments where Zookeeper is either not easy to deploy/manage or there are better alternatives to it there is a need to run Kafka with other metastore implementation than Zookeeper.

Etcd can provide the same semantics as Zookeeper for Kafka and since Etcd is the favourable choice in certain environments (e.g. Kubernetes) Kafka should be able to run with Etcd as well.

From the user's point of view should be straightforward to configure to use etcd by just simply specifying a connection string that point to etcd cluster.

Proposed Changes

This KIP proposes introducing the capability to store consensus and metadata for Brokers Etcd as well. Whether the metadata is stored in Etcd or Zookeeper is controlled through configuration passed to Kafka.

To avoid introducing instability in the first iteration the original interfaces should be kept and only the low level Zookeeper API calls should be replaced with Etcd API calls in case Kafka is configured to use Etcd.

*The methods of **ZooKeeperClient** used for storing metadata in Zookeeper are factored out into an interface called **KafkaMetastore** and **ZooKeeperClient** becomes an implementation of **KafkaMetastore** interface for Zookeeper. This way the metastore implementation is abstracted out.*

*Another implementation **EtcdClient** is added for Etcd.*

*Method parameters/constructor parameters of type **ZooKeeperClient** are changed to **KafkaMetastore**. Similarly fields and variables of type **ZooKeeperClient** are changed to type **KafkaMetastore**.*

The list of impacted classes:

- ***KafkaZkClient***

Library used for interacting with ETCD: com.coreos:jetcd-core:0.0.2.

Details of the change can be viewed from the github fork

New or Changed Public Interfaces

```
trait KafkaMetastore {

  def registerZNodeChangeHandler(zNodeChangeHandler: ZNodeChangeHandler): Unit
  def unregisterZNodeChangeHandler(path: String): Unit
  def registerZNodeChildChangeHandler(zNodeChildChangeHandler: ZNodeChildChangeHandler): Unit
  def unregisterZNodeChildChangeHandler(path: String): Unit
  def registerStateChangeHandler(stateChangeHandler: StateChangeHandler): Unit
  def unregisterStateChangeHandler(name: String): Unit
  def handleRequest[Req <: AsyncRequest](request: Req): Req#Response
  def handleRequests[Req <: AsyncRequest](requests: Seq[Req]): Seq[Req#Response]
  def waitUntilConnected(): Unit
  def sessionId: Long = ???
  def close(): Unit

}
```

Compatibility, Deprecation, and Migration Plan

- *Consensus and metadata storage systems should be compatible for Zookeeper-based implementation.*

Rejected Alternatives

- *None*