Kafka on Kubernetes without Zookeeper

Molnar Balint

BANZAICLOUD

Kafka on Kubernetes

Agenda

- Apache Kafka
- Zookeeper
- Kafka with Zookeeper
- Etcd
- Zookeeper vs Etcd
- Kafka with Zookeeper drawbacks
- Kafka with Etcd
- Kafka on Kubernetes
- Demo
- Pipeline
- QA

BANZAICLOUD

Kafka

- Apache Kafka is used for building real-time data pipelines and streaming apps.
  - Horizontal scalable
  - Fault-tolerant
  - Wicked fast
  - Requires Zookeeper

Zookeeper

- Centralized service for maintaining configuration
- Provides distributed synchronizations
- Reliable uses Zab consensus algorithm
- Part of the Hadoop ecosystem

Apache
Zookeeper

Kafka with Zookeeper

- Uses Zookeeper to store metadata
  - About broker's state
  - About Controller state
  - Relies on Zookeeper feature Ephemeral Node
  - For ACL it uses Persistent_Sequencial Node

Etcd

- Etcd is a distributed reliable key-value store for the most critical data of distributed system
  - Reliable, uses Raft consensus algorithm
  - Fast, 10,000 writes/sec
  - Kubernetes uses to store metadata
  - Can replace Zookeeper

Zookeeper vs Etcd

	etcd	Apache Zookeeper
Programming Language	Go	Java
Consensus Algorithm	Raft	ZAB(Zookeeper Atomic Broadcast)
Trusted by	Kubernetes	Hadoop Ecosystem
Maintainability	Easy	Hard

Kafka with Zookeeper drawbacks

- Kafka with Zookeeper drawbacks:
  - Kafka has a vibrant community, but Zookeeper...
  - Maintaining Zookeeper is painful, lot of tweet/discussion states Kafka is awesome but why Zookeeper?
  - It has an early KIP which tried to make metadata store plugable (KIP-30)

Kafka with Etcd

- Kafka with Etcd:
  - https://github.com/banzaicloud/apache-kafka-on-k8s
  - Synchronous ZkClient ⇒ Async ZkClient
    - https://issues.apache.org/jira/browse/KAFKA-5027
  - Jetcd (java etcd client)
  - Zetcd (Zookeeper Etcd Proxy)

Kafka with Etcd

- Ephemeral Node in Etcd
  - This concept does not exists in Etcd, but doable with Lease
    - TTL, needs to be renewed otherwise the key/value gets deleted
- Persistent Sequential Node in Etcd
  - Etcd can use prefix during getting a key and sort the result(descending)
  - Using transactions to check if anyone modified the key during the update

Kafka with Etcd

```
// get the last sequential key
val seqKey = getKey(sortOrder(descending), limit(1), withPrefix(key))

// determine next sequence number
seqKey match {
    case SEQKEYREGEX => seqNumber + 1
    case error => _
}.getOrElse(0) // if no <key> found than default sequence number to 0

// if __<key> hasn't been created yet OR
// modification revision of __<key> hasn't been changed by someone else
If(Key not created) || (Key not modified)
    EarlierKeyValue = ""
    SetNewKey = value
```

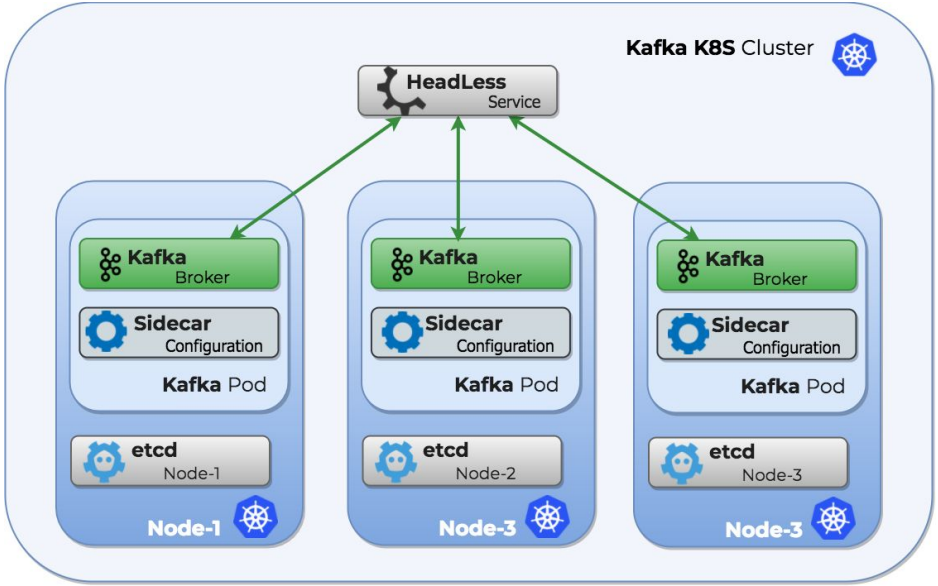
BANZAICLOUD

Kafka with Etcd

- Handling Children Nodes in Etcd
  - Prefixing children nodes (/parentkeyprefix/chilrdrenkey)
  - Getting back multiple results with range feature
  - Watcher's can be reused

Kafka with Kubernetes

- StatefulSet
  - The workload API object used to manage stateful apps
  - Provides ordering and uniqueness of the managed pods
  - Require *headless service* for the network identity of the pods.



APACHE kafka

etcd

BANZAICLOUD

Future work

- https://issues.apache.org/jira/browse/KAFKA-6598
  - We need to create a KIP, we want to contribute this back to the official Kafka repo
- Make Kafka available outside of the Kubernetes cluster

Demo

BANZAICLOUD

Pipeline

A Next Generation **Heroku** / **Cloud Foundry** like PaaS on steroids.

A full managed runtime for enterprise software.

Pipeline

https://github.com/banzaicloud/pipeline


Do you have any
questions?