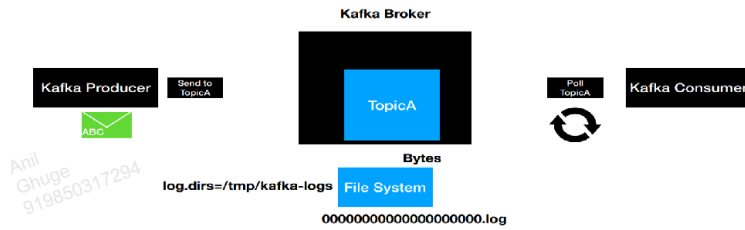# Apache Kafka

Commit Logs

Lets we have Kafka broker, topic, producer and consumer. when the producer sends the message it first reaches to the topic and then immediately the record written to a file system in the machine where the kafka broker is installed.

Kafka Broker

- ✓ The record is always written to the file system in the form of bytes.
- ✓ The file system where the files are writing has been configured the property **log.dirs of the server.properties** file.

```
server.properties  ×
C: > kafka 2.13-3.1.0 > config > server.properties
  51    socket.receive.buffer.bytes=102400
  52
  53    # The maximum size of a request that the socket server will accept (protect
  54    socket.request.max.bytes=104857600
  55
  56
  57    ############################# Log Basics #############################
  58
  59    # A comma separated list of directories under which to store log files
  60    log.dirs=/tmp/kafka-logs
  61
```

- ✓ The file will be created with an extension of .log

Each Partition will have its own log, meaning if we have 5 partitions then we have 5 log files created in the file system that's why these logs are called as partition commit logs.

To View the Commit Logs

kafka-run-class.bat kafka.tools.DumpLogSegments --deep-iteration --files /tmp/kafka-logs/test-topic-0/00000000000000000000.log

After the record has been written to the log file, that's then the records were produced will be committed. so when the consumer who is continuously polling for the records can only see the records that are committed to the file system. As the new records are producing the topic, then the new records are appended to the log file and then process contagiously.

Apache Kafka

Retention Policy

- ✓ Retention policy is one of the key property that is going to determine how the , how long the messages going to be retained.

- ✓ Retention policy is configured using the property log.retention.hours in server.properties file.

```
server.properties  ×
C: > kafka_2.13-3.1.0 > config > server.properties
  94
  95    ############################# Log Retention Policy #############################
  96
  97    # The following configurations control the disposal of log segments. The policy can
  98    # be set to delete segments after a period of time, or after a given size has accumulated.
  99    # A segment will be deleted whenever *either* of these criteria are met. Deletion always happens
 100    # from the end of the log.
 101
 102    # The minimum age of a log file to be eligible for deletion due to age
 103    log.retention.hours=168
 104
 105    # A size-based retention policy for logs. Segments are pruned from the log unless the remaining
 106    # segments drop below log.retention.bytes. Functions independently of log.retention.hours.
 107    #log.retention.bytes=1073741824
 108
 109    # The maximum size of a log segment file. When this size is reached a new log segment will be created.
 110    log.segment.bytes=1073741824
 111
 112    # The interval at which log segments are checked to see if they can be deleted according
 113    # to the retention policies
 114    log.retention.check.interval.ms=300000
 115
```

default retention period is 168 hours(7 days)