1. Text

   Description automatically generated  
   
2. So we learnt about the internal working of Producer API.
3. Before we start creating more producers, let me summarize the producer internals.
4. We use **KafkaProducer.send(ProducerRecord)** to hand over the **ProducerRecord** (having msg) to the KafkaProducer.
5. KafkaProducer will internally **serialize** the **key** and **value** of **ProducerRecord**.
6. Then the KafkaProducer will determine the target **partition** to deliver the msg.
   1. We can provide custom partitioner class using Properties object (Config)
   2. or use a key of ProducerRecord and let the KafkaProducer use default partitioner.
7. The serialized msg will sit in the buffer for the specific partition of a specific topic. We have separate buffer for each partition of each topic.
8. Finally, the I/O thread running in background will pick up some msgs from a buffer and combine them to make one single data packet and send it to the Broker.
9. Broker will save the data in the log file and returns an acknowledgement to the I/O thread.
10. If the I/O thread doesn’t receive the acknowledgement, it will retry sending the packet and again waits for acknowledgement.
    1. If I/O thread doesn’t receive the acknowledgement after some retries or get an error msg, the I/O thread will give error to the KafkaProducer.send(ProducerRecord) method.
11. Diagram

    Description automatically generated