Kafka Fundamentals Brain Teasers: Key Takeaways

Key Kafka Takeaways

From each of these scenarios, here are some of the key lessons that we learn about Kafka:

- **A partition contains messages from one topic; it is a subset of the messages in a topic.** Partitions are physically co-located groups of messages. Topics are logical groupings of messages, generally spread across multiple partitions.
- **Replication factor is the total number of instances of a partition on the Kafka cluster**, not the number of copies. The replication factor is one more than the number of followers.
- **Kafka is a pull architecture**, not a push architecture. Kafka doesn't tell consumers that new messages are available; consumers ask Kafka for messages. Leaders don't push new messages to followers; followers periodically ask leaders if new messages are available.
- **The consumer offset identifies the message to be read next**, not the message last read. Think of an offset like a bookmark. We typically put the bookmark at the next page we're going to read, not the one we just read.
- Along this vein, **what is reported as "log end offset" is actually one more than the offset of the last written message**. You'll see this both in the Kafka CLI and Confluent Control Center's views of topics.
- **Keys of Kafka messages are not to be confused with primary keys in databases**; they do not uniquely identify messages, but rather they allow for semantic partitioning. In Kafka, many messages may have the same key.
- While we use keys with the hope of physically co-locating same-key messages in the same partition, **it is not always guaranteed that all messages with the same key are stored on the same partition**. The default partitioning strategy selects the index of the partition as `hash(key) % (number of partitions)`. Hashing will always give the same output for a given input. But if the number of partitions we divide by changes, generally the remainder does too.
- **Deletion works on entire segments**, not individual messages. This means that if you leave the default retention period at seven days, messages that are eight days old could still be left around. The newest message in a segment must be older than the retention period for the entire segment to be deleted. If even a single message in the segment is not older than the retention period, the whole segment sticks around.
- **The active log segment is not considered during compaction.** Consider the active segment specially, both for this reason and because new messages might have arrived since the last time the cleaner ran.