Changes in Kafka 0.8

The primary goal of Kafka 0.8 is to add the intra-cluster replication support. Before 0.8, if a broker goes down, the data on that broker is not available until the broker comes back. This is a severe limitation for some applications. By adding replication in 0.8, we increase both the availability and the durability of Kafka. The details of the replication design can be found here and here. The following are the replication related changes in 0.8:

- New administrative tools for creating, listing and deleting topics.
- New zookeeper data structures for storing replica assignments and state information about each partition (e.g., which replica is the leader and which replicas are in-sync with the leader).
- An embedded controller for detecting broker failures and managing the state change of each partition (e.g., moving a partition from offline to online and changing the leader of a partition when the previous leader fails).
- Modified RPC protocol to allow a producer to wait for an ack until a message reaches a certain # of replicas and new RPC requests for clients to get the metadata of each topic (e.g., # of partitions, the leader of each partition), and for the controller to communicate with the broker.
- An async processing framework for handling produce and fetch requests that may not be satisfied immediately.
- A replica manager that manages the set of replicas assigned to a broker.
- A significant system test framework for testing various failure scenarios.
- A migration tool for mirroring pre-0.8 data to 0.8.

Because of the significant changes in zookeeper data and the RPC protocol, we decided to make 0.8 a non-backward compatible release. Therefore, we try to combine a few other non-backward compatible changes in this release to make future releases easier.

- New RPC protocol for all types of requests.
- A new format of the log that makes the offset of each message logical, instead of physical.
- Standardized config names.
- Exposing all jmx metrics using the metrics core package.