

KIP-487: Client-side Automatic Topic Creation on Producer

- [Status](#)
- [Motivation](#)
- [Public Interfaces](#)
- [Proposed Changes](#)
- [Compatibility, Deprecation, and Migration Plan](#)
- [Rejected Alternatives](#)

Status

Current state: *Under Discussion*

Discussion thread: [here](#)

JIRA: [here](#)

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Motivation

Kafka has a feature that allows for the auto-creation of topics. Upon a produce or consume request, when fetching metadata for a topic that does not yet exist, the topic may be created if the broker enables topic auto creation. In the early days of Kafka, this was one of only a few ways to create topics. However, with the addition of AdminClient, this functionality is no longer the recommended way to create topics.

There are a few reasons the current implementation is not optimal. Metadata requests should not modify the metadata, or perform any other tasks besides obtaining the metadata. In addition, a consume request should not necessarily create a topic, since a consumer can retry until the topic is created and can actually consume.

However, many cloud users rely on the auto-create functionality. They often use software that they did not write and can not easily change that relies on auto-creation. Thus, moving the auto-create functionality to the producer will still support topic auto-creation, but will do away with some of the negative aspects of the current implementation. Auto-creation can then be configured on a client-by-client basis.

Public Interfaces

The producer will have a new configuration added.

Name	Description	Type	Default	Valid Values	Importance
producer. auto. create. topics	Configures how topic auto-creation will occur. 'False' does allow auto-topic creation. 'Server-side' allows the server to create the topic if the broker's 'auto.create.topics.enable' is true. 'Client-side' creates the topic client-side. Client-side auto-creation is only supported by brokers with version greater than 0.10.1.1.	String	allow- server- side	[false, allow- server- side, client- side]	medium

The goal is to deprecate automatic topic creation on the broker. At some point in the future, it may be useful to set the broker's auto.create.topics.enable default to false. However, for compatibility reasons, doing so is not possible at this time. Likewise, it would make sense to eventually change the default of the consumer's allow.auto.create.topics to false since it has no effect if the broker configuration is false.

Proposed Changes

The idea is to move the configuration and ability to automatically create topics client-side on the producer. Broker auto-topic creation will be deprecated. This includes deprecating the configuration 'auto.create.topics.enable' on the broker and 'allow.auto.create.topics' on the consumer. In the producer, auto-creation of a topic will occur through a specific request rather than through a side effect of requesting metadata. Unlike the configuration in the consumer, the ability to automatically create topics client-side will be independent of the server-side automatic topic creation controlled by the broker configuration.

If the broker config, 'auto.create.topics.enable' is set to true, a warning will be given that states the config is deprecated.

If server-side auto-creation is enabled with the broker's auto.create.topics.enable, the broker's defaults for partitions and replication factor will be used. When client-side auto-creation is enabled, broker-side defaults will be used if the broker supports KIP 464. Client-side auto-creation is not supported for older brokers.

Compatibility, Deprecation, and Migration Plan

This KIP argues for deprecation of the broker config `auto.create.topics.enable`. However, the default will remain true, so there will be no compatibility issues with current usage of the config and its functionality.

In order to automatically create topics with the producer, `producer.auto.create.topics` config must be set to 'client-side.' If the broker does not support KIP 464, `auto.create.num.partitions` and `auto.create.replication.factor` must be specified.

This KIP does not plan on changing the broker default. However, it is useful to understand compatibility issues if this were to happen.

- By default, 'allow-server-side' auto-creation is used. The broker's 'auto.create.topics.enable' must be configured as true.
 - Although the broker config and server-side auto-creation is deprecated, this is the default to ensure compatibility.
 - If the broker config is set to false, the topic will not be created automatically.
- Systems that rely on auto-creation would be encouraged to set the producer config `producer.auto.create.topics` to 'client-side.'
 - This will effectively override the deprecated broker config, so even if it is configured to true, the server will not auto-create topics.
 - Systems with older broker versions that do not support KIP-464 would not be compatible with this option.
- Systems that want to turn off auto-creation completely should set the producer config to false. This will override the broker config for this producer.
 - This will fulfill the same role as 'allow.auto.create.topics' on the consumer, but for the producer.
- Systems that utilize auto-creation on the consumer will no longer be able to do so with only the producer config enabled.
 - Enabling the broker config or making further changes to the code is required.

Rejected Alternatives

- Enabling true by default on the producer config:
 - Topic auto-creation is a feature that is not supported as it once was, so defaulting to false supports this trend towards using admin client to create topics. Auto-creation does not work on older brokers without configuration client-side anyway, so it might be more clear that configuration must be done in order to keep using this feature.
- Add configurations producer-side for replication factor, partitions of new topics
 - The idea is to keep this simple; configuring these values is not simple and could add to more issues for users.