

KIP-246: Connect producers and consumers should inherit worker configs

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

This page is meant as a template for writing a KIP. To create a KIP choose Tools->Copy on this page and modify with your content and replace the heading with the next KIP number and a description of your issue. Replace anything in italics with your own description.

Status

Current state: *Withdrawn*

Discussion thread: [here](#) [Change the link from the KIP proposal email archive to your own email thread]

JIRA: [KAFKA-6387](#)

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

Motivation

The Connect worker configuration must specify the set of configuration properties for the worker's consumers and producers used to read and write to the internal topics, and except for the basic broker connection information the configuration properties for Connect's producers and consumers that are used for source and sink connectors. Currently, the configuration between the worker's internal producers and consumers and the producers and consumers used for connectors are unrelated. To simplify configuration in some situations, the producer and consumer used for connectors should inherit from the worker-level configuration properties.

Public Interfaces

No public interfaces (code or configuration definitions) will change.

Proposed Changes

Connect worker configurations define connection properties for the following types of connections being made to the Kafka cluster:

1. the worker group membership and producers/consumers for internal topics, with no prefix
2. producers for source connectors, prefixed with "producer." (e.g., "producer.retries=1").
3. consumers for sink connectors, prefixed with "consumer." (e.g., "consumer.max.partition.fetch.bytes=10485760").

Currently, any top-level configurations are set for the internal producer/consumers do not affect the producers and consumers used for connectors. So, if the same configuration is to be overridden for all consumers/producers, the configuration must be set in multiple places.

This proposal suggests changing the behavior so that the configuration for the producer and consumer used for connectors (items 2 and 3 in the list above) *inherits* any configurations defined at the top-level (item 1 in the list above). The `producer.*` and `consumer.*` properties would override any inherited properties, resulting in exactly the same behavior as today.

Compatibility, Deprecation, and Migration Plan

NOTE: After further evaluation, there are a number of cases where this behavior might actually change how the producers and consumers used for connectors are configured. Consider a worker configuration includes the following configuration properties:

```
retries=2
max.partition.fetch.bytes=262144
```

This configuration file does not define "consumer.max.partition.fetch.bytes" or "producer.max.partition.fetch.bytes", so currently these would default to 1048576. However, after this proposed change if implemented the "consumer.max.partition.fetch.bytes" and "producer.max.partition.fetch.bytes" values would be the inherited value of 262144, not the default 1048576.

In short, implementing this change would break backward compatibility. We could implement a configuration switch that controls whether the configurations are inherited, but this adds complexity to the already-complex configuration mechanism. Therefore, withdrawing this KIP.

Rejected Alternatives

Using a new configuration property to control whether the configuration properties are inherited would work, but would also add unnecessary complexity.