KIP-306: Configuration for Delaying Response to Failed Client Authentication

Status

Current state: *Approved*

Discussion thread: *here*

JIRA: KAFKA-6950

Release: 2.1.0

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

Motivation

Kafka network threads are mainly responsible for establishing connections with clients and other brokers, performing authentication with them, receiving Kafka protocol requests and responses, and for sending out Kafka protocol requests and responses. Performing authentication is one of the most expensive operation performed in the network thread, so much so that it could end up saturating them, preventing from getting any useful work from being done. Given this, a misconfigured application trying to connect with incorrect (or stale) credentials could cause a denial of service like situation. To help mitigate this issue, this KIP proposes to introduce a way of delaying response to a failed authentication.

Public Interfaces

This KIP proposes to add a new broker configuration `connection.failed.authentication.delay.ms` which specifies the amount of time (in milliseconds) to delay a failed authentication response and subsequent socket disconnection by. In general, this must be configured to be lower than `request.timeout.ms`.

Configuration Name: `connection.failed.authentication.delay.ms`
Type: INT
Explanation: This configuration specifies the amount of time in milliseconds by which a response to a failed authentication will be delayed by.
Default Value: 100
Priority: LOW

Proposed Changes

The proposal is to delay sending out the "authentication failed" response along with any associated clean up work like socket disconnection, etc. The implementation will be similar to `ClientQuotaManager#delayQueue` and `ClientQuotaManager#ThrottledRequestReaper` mechanism.

Compatibility, Deprecation, and Migration Plan

The change in this KIP is backward compatible as it does not change the protocol in any way. The only difference is that clients will see a 100ms delay on failed authentications.

Rejected Alternatives

None.