

# KIP-764: Configurable backlog size for creating Acceptor

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

## Status

**Current state:** "Accepted"

**Discussion thread:** [here](#)

JIRA:

 Unable to render Jira issues macro, execution error.

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


## Motivation

In large scale Kafka cluster which handles requests from massive clients, preferred leader election (e.g. upon restarting broker) could cause many clients to open connection to a broker in a short period.

Sometimes this causes Acceptor socket's SYN backlog to be filled up. In case this happens, further incoming connections will be handled differently depending on `tcp_syncookies` kernel parameter in Linux.

1. Drop further SYN packets (`tcp_syncookies = 0`)
  - Typically this should not be a critical problem since clients will attempt reconnecting (depending on `tcp_syn_retries` though)
  - However, retries will cause certain delay until successful connection so should be avoided as far as possible
2. SYN packets are handled with "SYN cookies" (`tcp_syncookies = 1`)
  - In short, SYN cookies is a stateless way to handle SYN without consuming SYN backlog
  - It's known that this could cause subtle bug that producer slowing down due to inconsistent window-scaling factor between client and broker

◦ Please refer

 Unable to render Jira issues macro, execution error.

's comment for the detailed explanation

about this issue

Both are undesirable, and can be mitigated by increasing backlog size passed to `ServerSocket#bind()` as necessary.

## Public Interfaces

We propose a new KafkaConfig

- **socket.listen.backlog.size**
  - This may be an integer config and passed as the parameter for `ServerSocket#bind()`
    - <https://github.com/apache/kafka/blob/2.8.0/core/src/main/scala/kafka/network/SocketServer.scala#L663>

## Proposed Changes

### KafkaConfig

- Add new integer integer config **socket.listen.backlog.size** with default value 50

## SocketServer

- Pass **socket.listen.backlog.size** to [ServerSocket#bind\(\)](#) when creating Acceptor

## Compatibility, Deprecation, and Migration Plan

- No impact

## Rejected Alternatives

1. Increase static backlog size without introducing new config
  - Increasing backlog size may consume more memory, so appropriate value depends on the environment