**Assumptions**

1. The project is designed keeping threads in mind and not for processes.

**Usecase**

A single shared channel acts like a multi producer single consumer where different threads can push packets on to channel, the channel process(consume) the packets and send them to the server.

Each Single shared channel is equipped with IP address and port number it directs all the packets to.

The channel is limited with message rate per second capacity which indicates the rate at which the channel can process the packets. When it crosses the threshold, it pushes the packets onto ring buffer for future processing.

Ring buffer is a multi-producer single consumer lock free queue internally uses a synchronized circular buffer and can hold ‘n’ number of packets.

Channel parallelly handles the packets in the queue and incoming requests, used spin lock kind of technique to implement ring buffer.

Used two threads in single shared channel 1. To monitor message rate 2. To process packets in the queue.

**Usage**

1. Create/request a single shared channel by providing IP and port number
2. Send the packets through the newly created channel
3. Disconnect it from any of the thread.