Getting Kafka to work using Docker, and across multiple machines is not trivial. In this directory are several files that should allow Kafka to stream across machines, and while running zookeeper and the kafka broker in a Docker container.

* Make sure the firewall permits inbound/outbound messages to all machines.
* Run docker-compose up -d in WSL2 (for windows machines) to start zookeeper and the kafka broker. Note, WSL2 should be configured so that WSL2 is using the same IP address as the host
* Next, on the new machine, you can run kafka\_producer\_test.py and kafka\_consumer\_test.py in a docker container, and you should see results. You must make sure the IP address in these files is the same as the IP address as the machine running the broker. Also, you will need to do a pip install kafka-python to install the proper python package
* On the same machine as the broker, if you enter a container (that was also started with docker compose), and run the kafka\_\*\_test\_broker\_machine.py files, these will all work.

Now, we have full kafka communication between (1) the local machine (WSL), (2) Docker containers running on the local machine, and (3) Docker containers running on remote machines!

The following blog posts help demonstrate why doing this is hard, and understand what needs to be done to make it work:

<https://www.confluent.io/blog/kafka-listeners-explained/>

<https://www.confluent.io/blog/kafka-client-cannot-connect-to-broker-on-aws-on-docker-etc/>