Network Assignment 4

2014313303

홍태하

개발환경 : python 3.7.0, window 10에서 테스트 함.

1. 구현 방식
2. Receiver

Receiver는 code시작시에 packet loss probability를 사용자로부터 입력 받고, socket receive buffer size를 화면에 띄운다. 만약 그 값이 10MB보다 작다면 10MB로 바꿔준다.

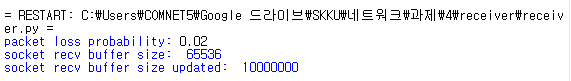
Socket을 10080port로 bind해주고 난 후 sender로부터 file 이름과 packet의 길이를 받는다. Packet 길이는 sender로부터 받을 packet의 총 개수를 의미한다. Pkt\_arr라는 이름의 이중리스트를 만들어서 한쪽에는 packet data를 저장하고 한쪽에는 packet 번호를 저장한다. 만약 그 packet을 아직 받지 못했거나 drop되었다면 packet 번호를 -1로 지정한다.

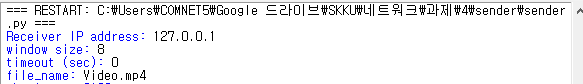
처음에 지정하였던 packet loss probability에 따라서 drop을 발생시키고 발생하지 않았다면 pkt\_arr에서 가장 큰 packet 번호를 sender쪽으로 보낸다. 그리고 sender 쪽에서 packet을 모두 전송하였다는 신호가 오면 받은 packet으로 file을 생성하고 종료한다.

1. Sender

Sender는 시작시에 Receiver IP address와 window size, timeout을 사용자로부터 입력 받는다. 그 후 receiver측으로 전달할 file name을 받는다. 이 file을 열어서 packet size(1400byte)만큼씩 나누어서 저장해 놓는다. 그리고 packet len과 file name을 receiver에게 보낸다. 그 후 receive\_ACK함수를 실행한다. Receiving\_ACK함수는 receiver로부터 ACK을 받는 함수이고 send\_pkt함수는 receiver쪽으로 pkt을 보내는 함수이다. Send\_pkt함수는 단순히 pkt number와 packet data를 보내는 작업을 하는데 pkt\_num은 최대 8자리 정수이다. 따라서 packet을 모두 보냈다면 모든 packet전송이 완료되었다는 의미로 99999999을 보내고 종료한다. 그렇지 않다면 pkt\_num과 packet data를 붙여서 전송한다. Receiving\_ACK함수는 window size만큼 pkt을 보내고 ACK이 왔을 때 packet이 drop 되었는지 확인하여 3 duplicated ACK을 한다. Pkt\_state라는 리스트를 만들어서 pkt이 정상적으로 보내졌다는 것이 확인되면 1로 그 값을 바꾸고 그렇지 않다면 0의 값을 가지게 된다. Pkt을 모두 보냈으면 sending\_pkt을 통해 receiver에서 전송이 끝났다는 것을 알리고 종료한다.

1. How to run sender and receiver programs.





Receiver.py 를 켜서 먼저 packet loss probability를 입력한다. 그 후 sender.py를 실행시켜 receiver IP address, window size, timeout, file\_name을 입력한다.

아래는 sender와 receiver의 log 파일 예제이다.

