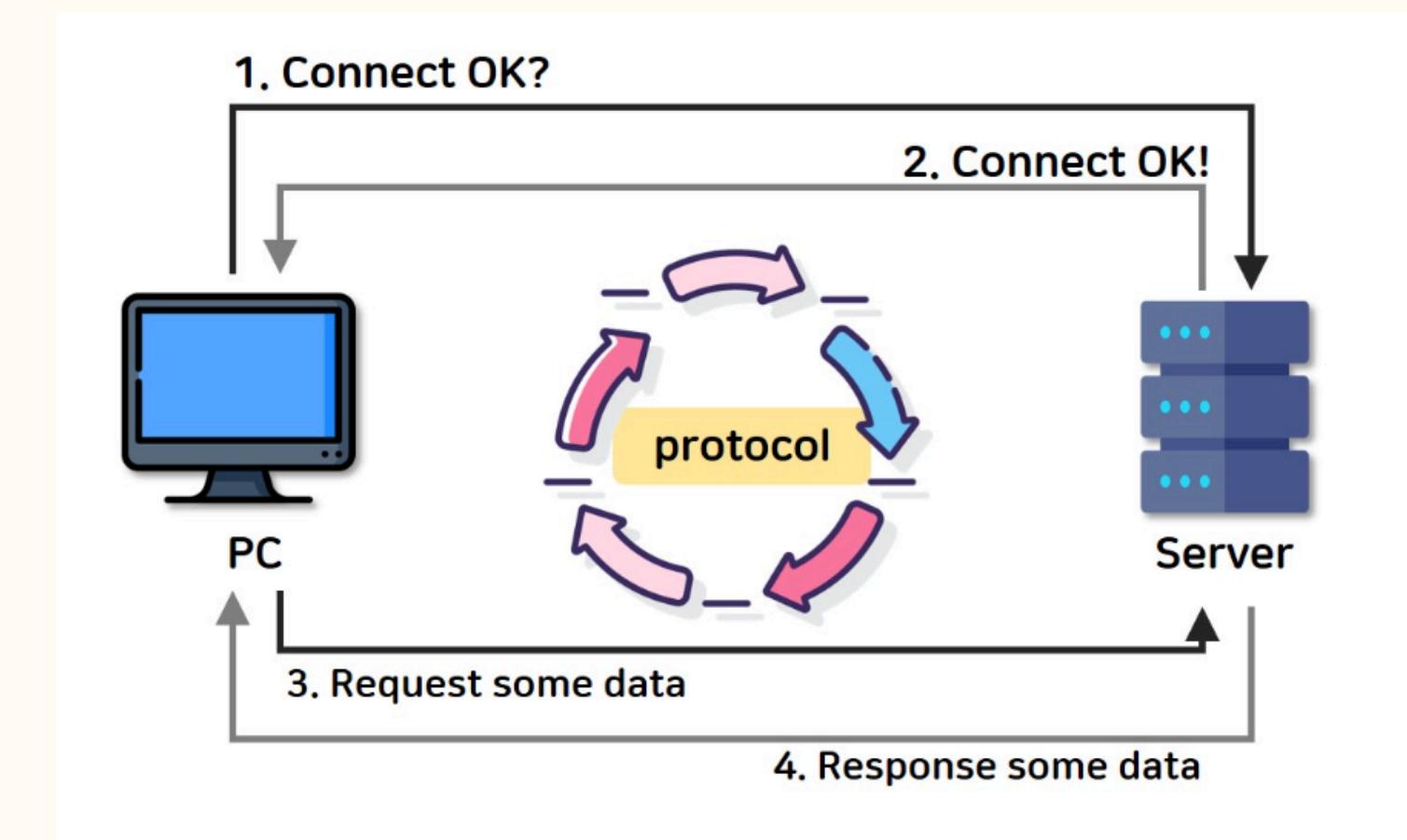


# T C P   v s   U D P

## 네트워크 전송 프로토콜 비교

# what is 'protocol'?

- 형식 (Format)?  
데이터가 어떤 구조로 담겨 있는지 (예: 헤더, 바디 구분)
- 순서 (Order)?  
누가 먼저 말하고, 어떻게 응답할지 (예: 요청 → 응답)
- 의미 (Meaning)?  
어떤 신호가 어떤 의미를 가지는지 (예: ACK → “데이터 받음”)



우린 어떤 규칙으로 데이터를 ‘전송’ 할래?

TCP? UDP?

## UDP

User Datagram Protocol

사용자 데이터그램 프로토콜

👉 “데이터그램”은 작은 데이터 조각(Packet)을 뜻

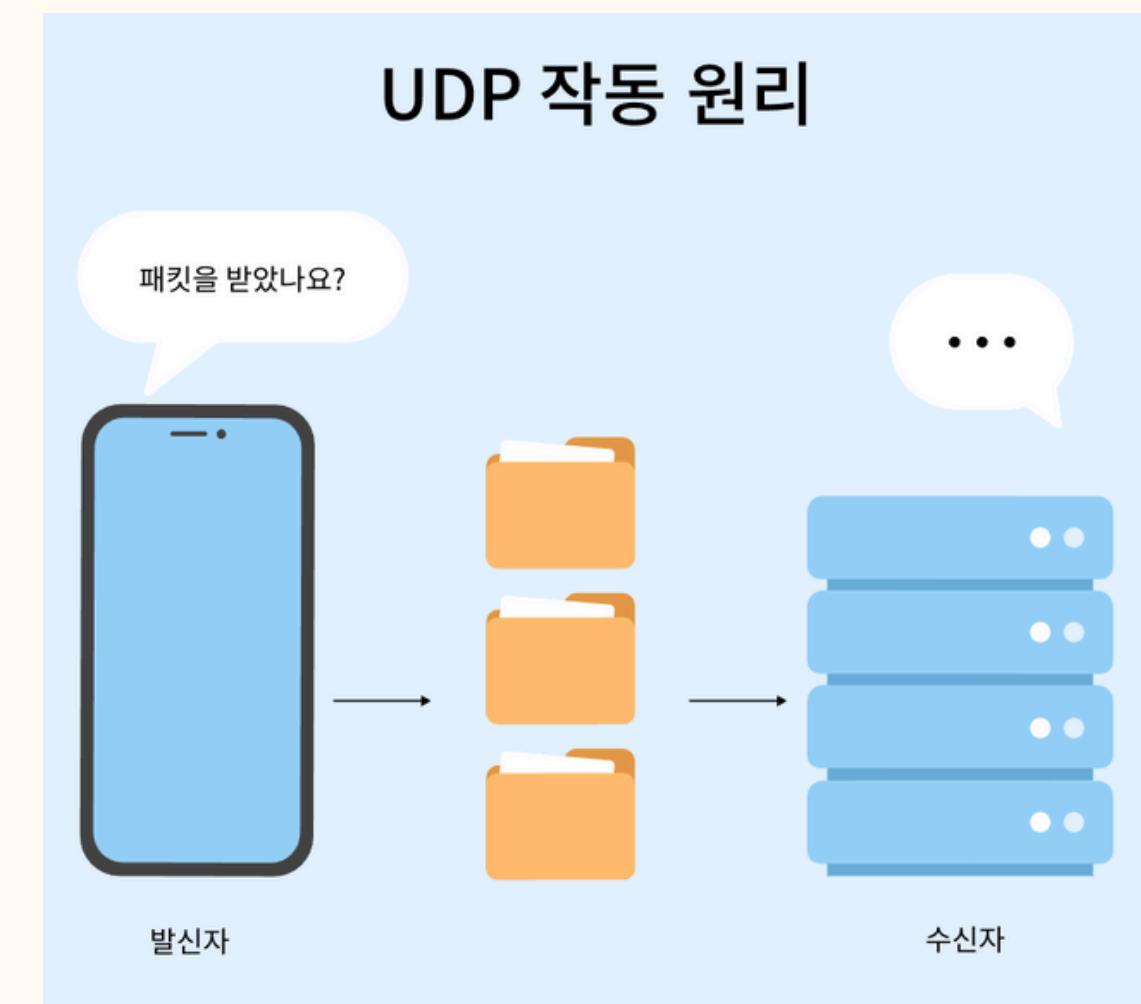
👉 즉, 사용자가 보낸 데이터를 조각 단위로 잘라서 네트워크에 “바로” 전송하는 프로토콜

## “비연결형(Connectionless)” 전송 프로토콜

데이터를 보내기 전에 상대방과 연결 협상(Handshake) 과정을 하지 않음

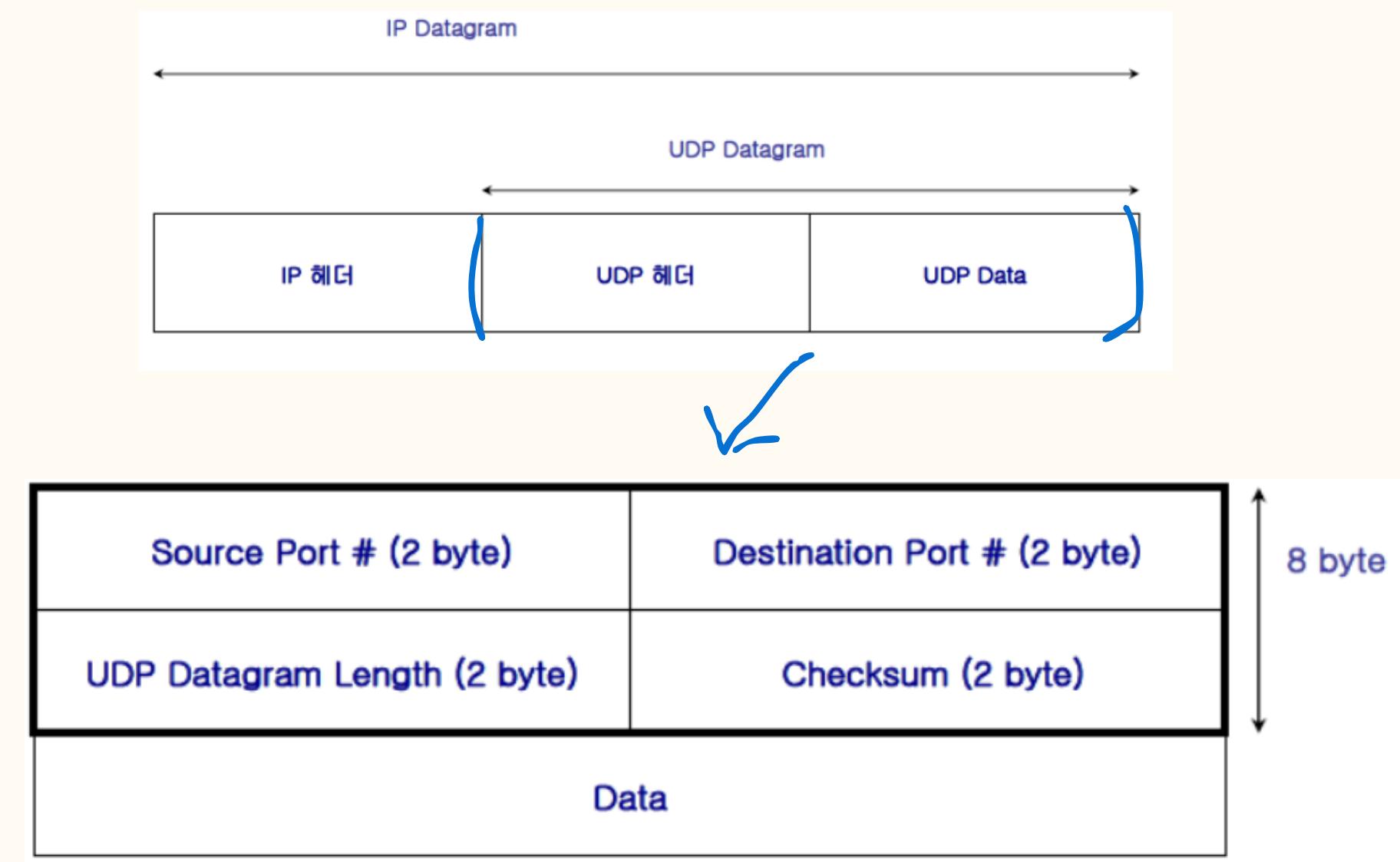
받은 쪽이 실제로 잘 받았는지 확인하지 않음 → 신뢰성 보장 X

대신 빠르고 단순하게 동작



팩킷 → 택배!

헤더 → 운송장!



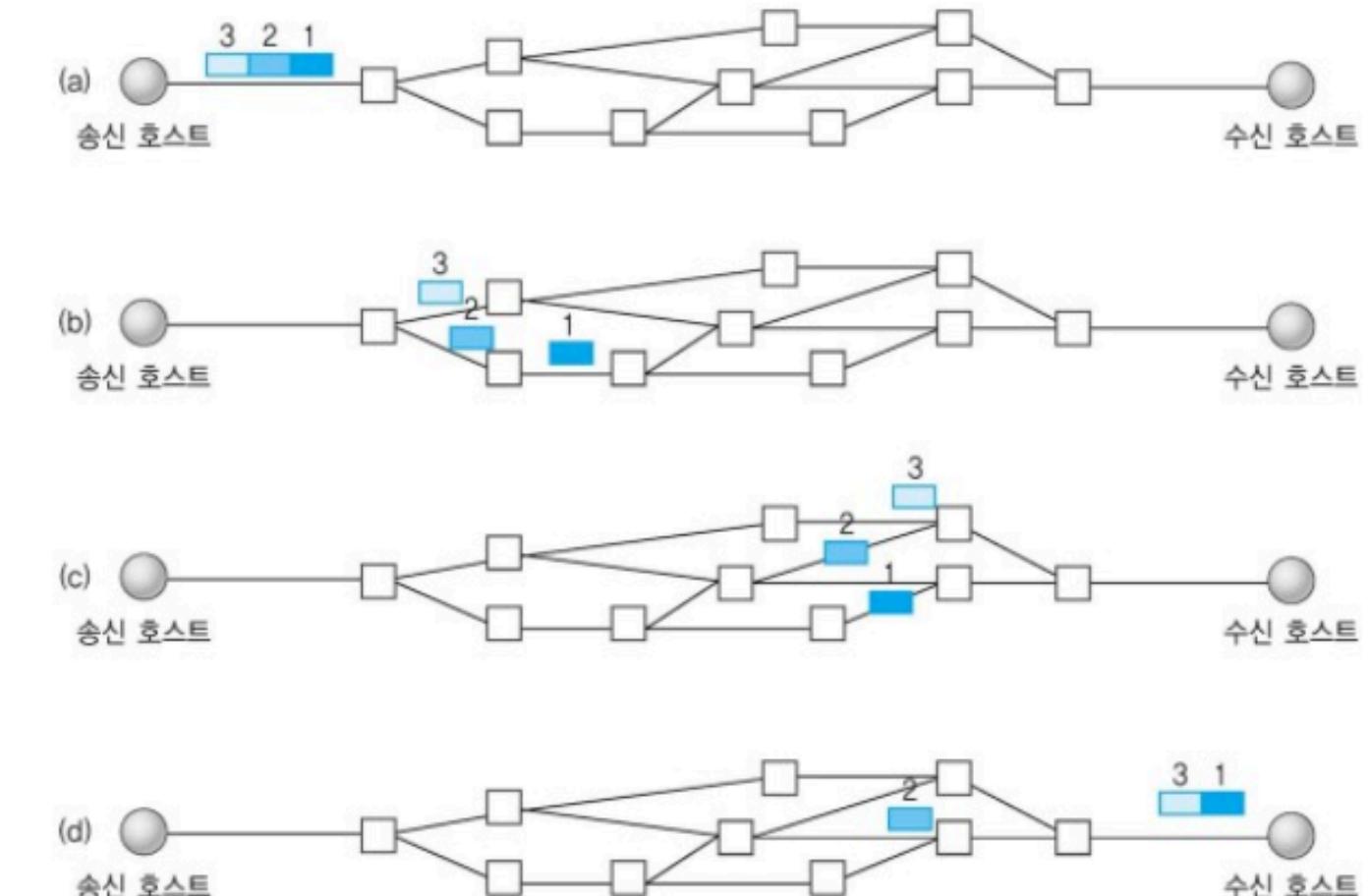
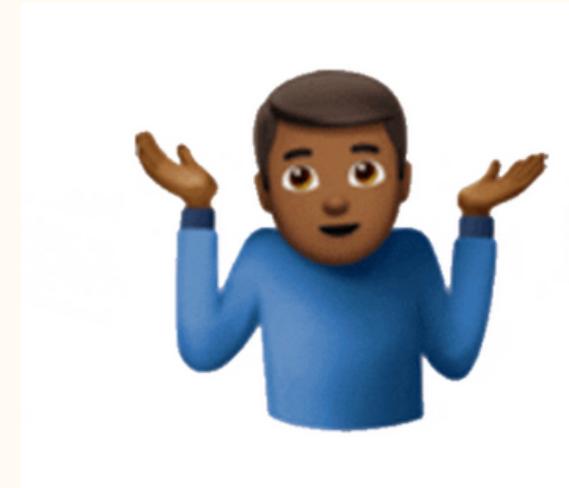
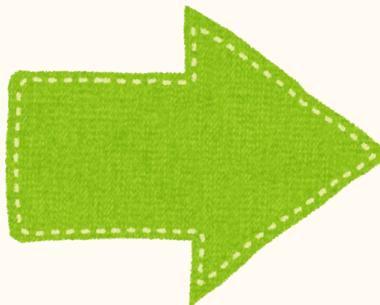
## 과정

**주소가 틀리면?**

**포트번호가 틀리면?**

**패킷이 꼬이면?**

**데이터가 손상되었으면?**



< 데이터그램 패킷 교환 방식 >

why?

**지연보다 즉각성이 중요한  
서비스에 적합**

**1:N, 전체 전송**



## TCP

Transmission Control Protocol

전송 제어 프로토콜

👉 TCP가 단순히 데이터를 보내는 게 아니라, 전송 과정을 “통제”

👉 즉, TCP는 데이터 전송이 빠짐없이, 순서대로, 안정적으로 이뤄지도록 관리·감독

(UDP의 보완형인 느낌이지만 사실 TCP가 먼저 등장하고 라이트형으로 UDP가 등장)

## “Connection-Oriented” 전송 프로토콜

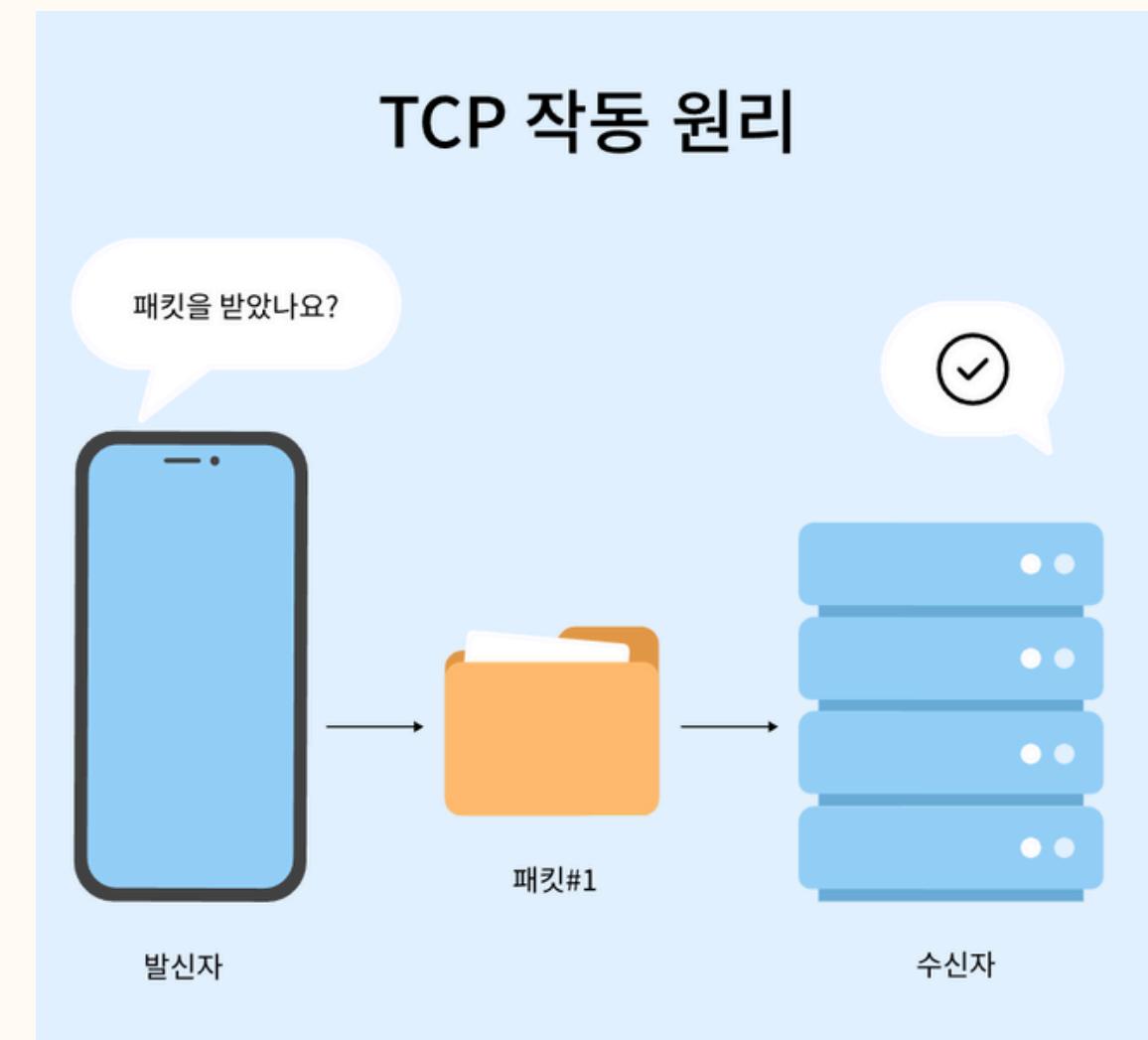
연결 관리 (Connection Management)

데이터 전송 보장 (Reliable Delivery)

흐름 제어 (Flow Control)

혼잡 제어 (Congestion Control)

TCP 작동 원리



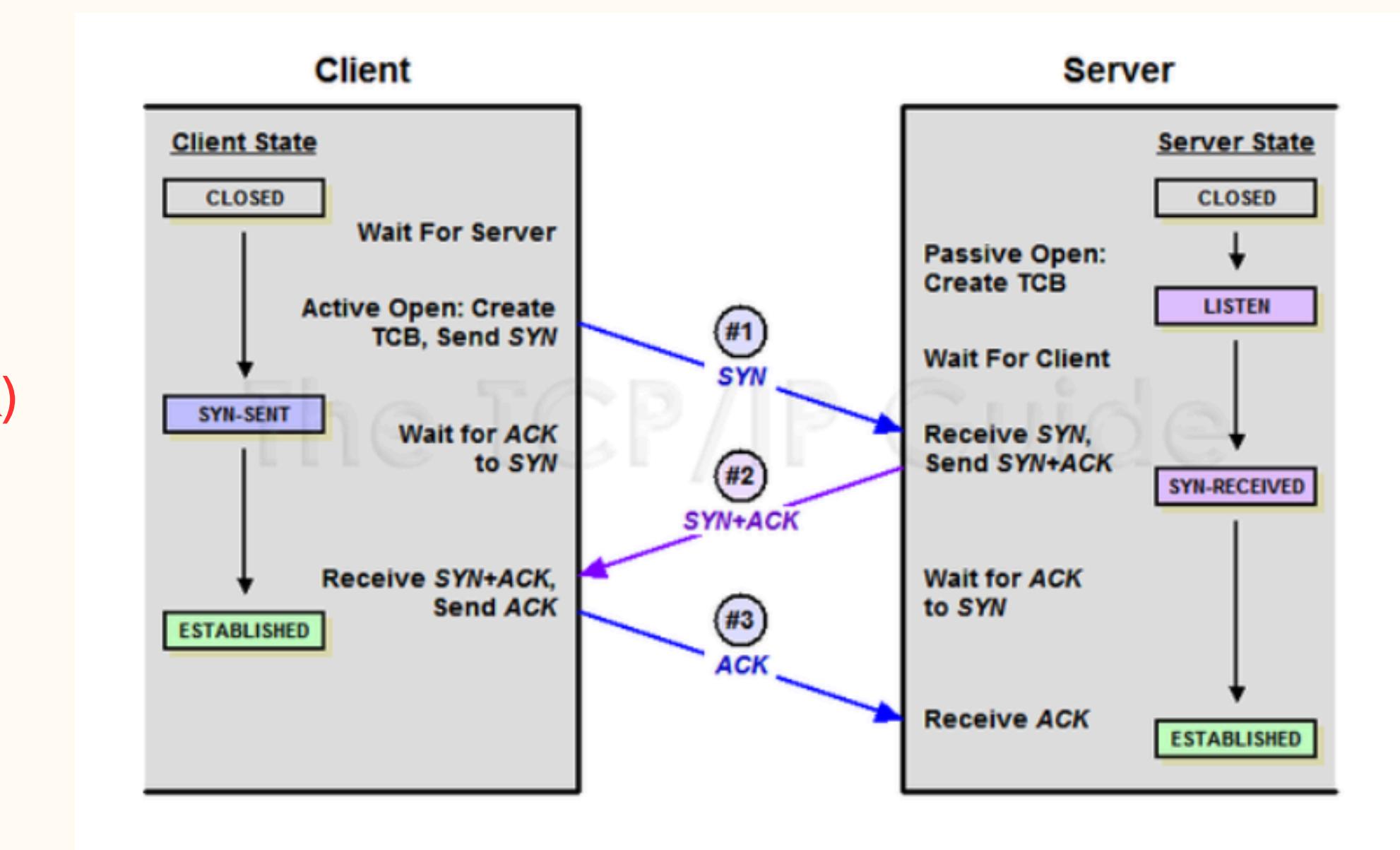
## 연결 관리 (Connection Management)

## 3 – way handshake

나: “여보세요, 들려요?” (SYN)

상대: “네, 들려요. 제 목소리도 들리죠?” (SYN+ACK)

나: “네, 잘 들려요!” (ACK)



## 연결 관리 (Connection Management)

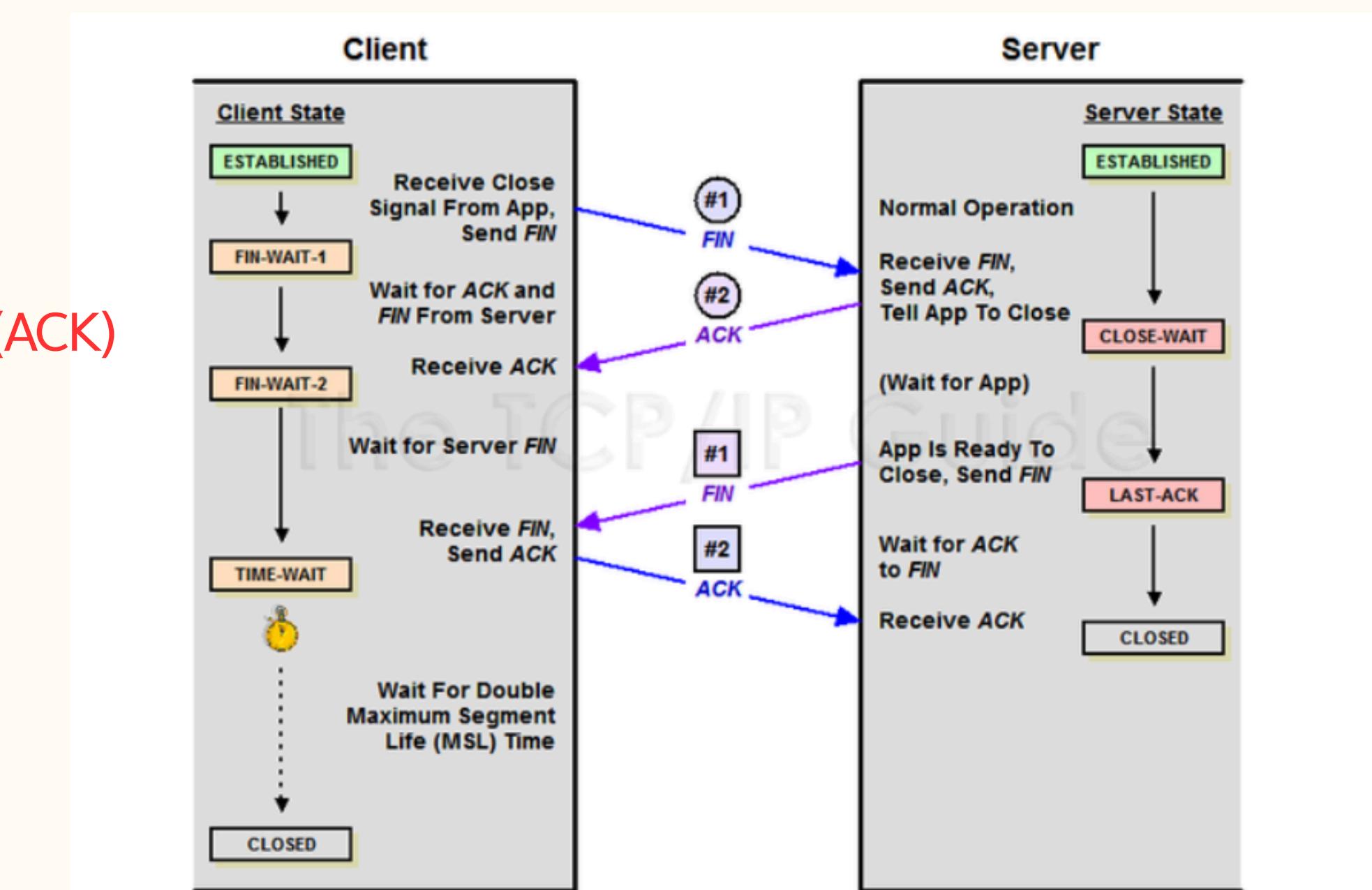
## 4 – way handshake

나: “나 할 말 다 했어, 끊을게” (FIN)

상대: “응, 알았어. 근데 나 아직 조금만 더 말하고 끊을게” (ACK)

상대: “나도 다 했어, 끊자” (FIN)

나: “응, 알았어” (ACK)



## 데이터 전송 보장 (Reliable Delivery)

### Sequence Number (순서 번호)

각 TCP 세그먼트(패킷)에 순서 번호 부여  
수신자가 “몇 번째 데이터인지” 확인 → 순서 꼬이면 재조립 가능

### 재전송 (Retransmission)

송신자가 보낸 세그먼트에 대해 일정 시간 안에  
ACK가 안 오면 → 손실로 판단  
같은 데이터를 다시 전송

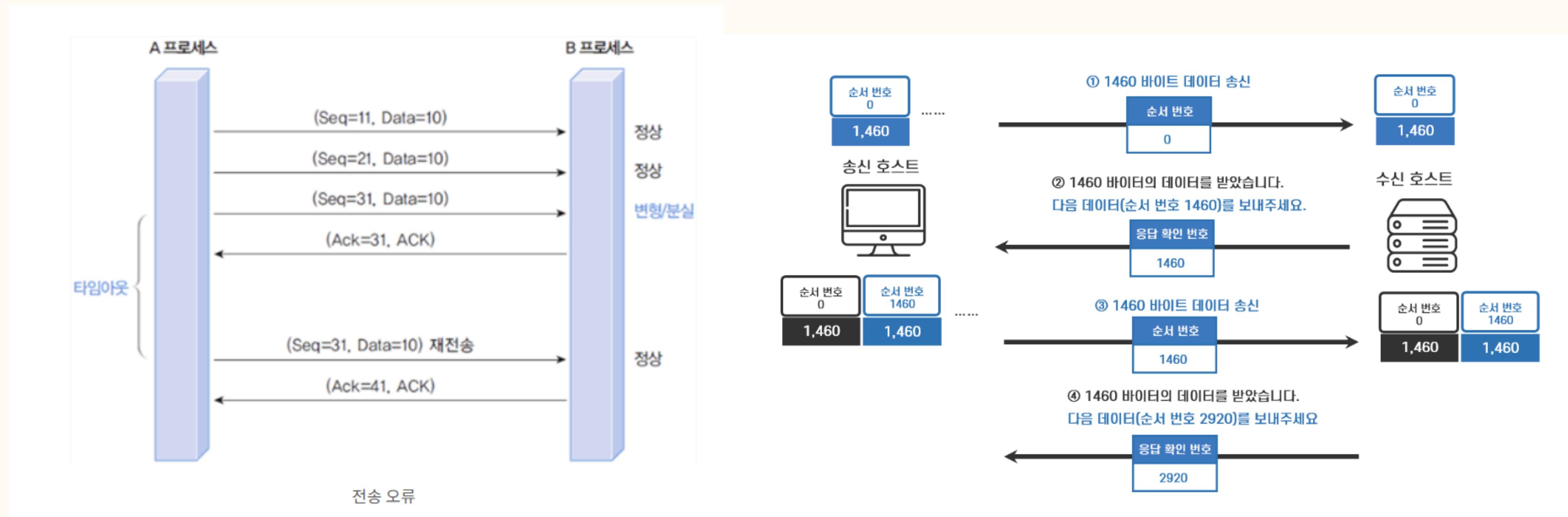
### Acknowledgment (ACK, 수신 확인 응답)

수신자가 정상적으로 받으면 ACK 응답 전송  
송신자는 ACK를 받으면 해당 데이터는 “성공적으로 전송됨”이라고 확정

### 체크섬 (Checksum)

TCP 헤더와 데이터에 대한 오류 검증 값 포함  
수신자가 체크섬 검사 → 손상된 세그먼트는 폐기  
송신자는 ACK가 오지 않으니 → 재전송

## 데이터 전송 보장 (Reliable Delivery)



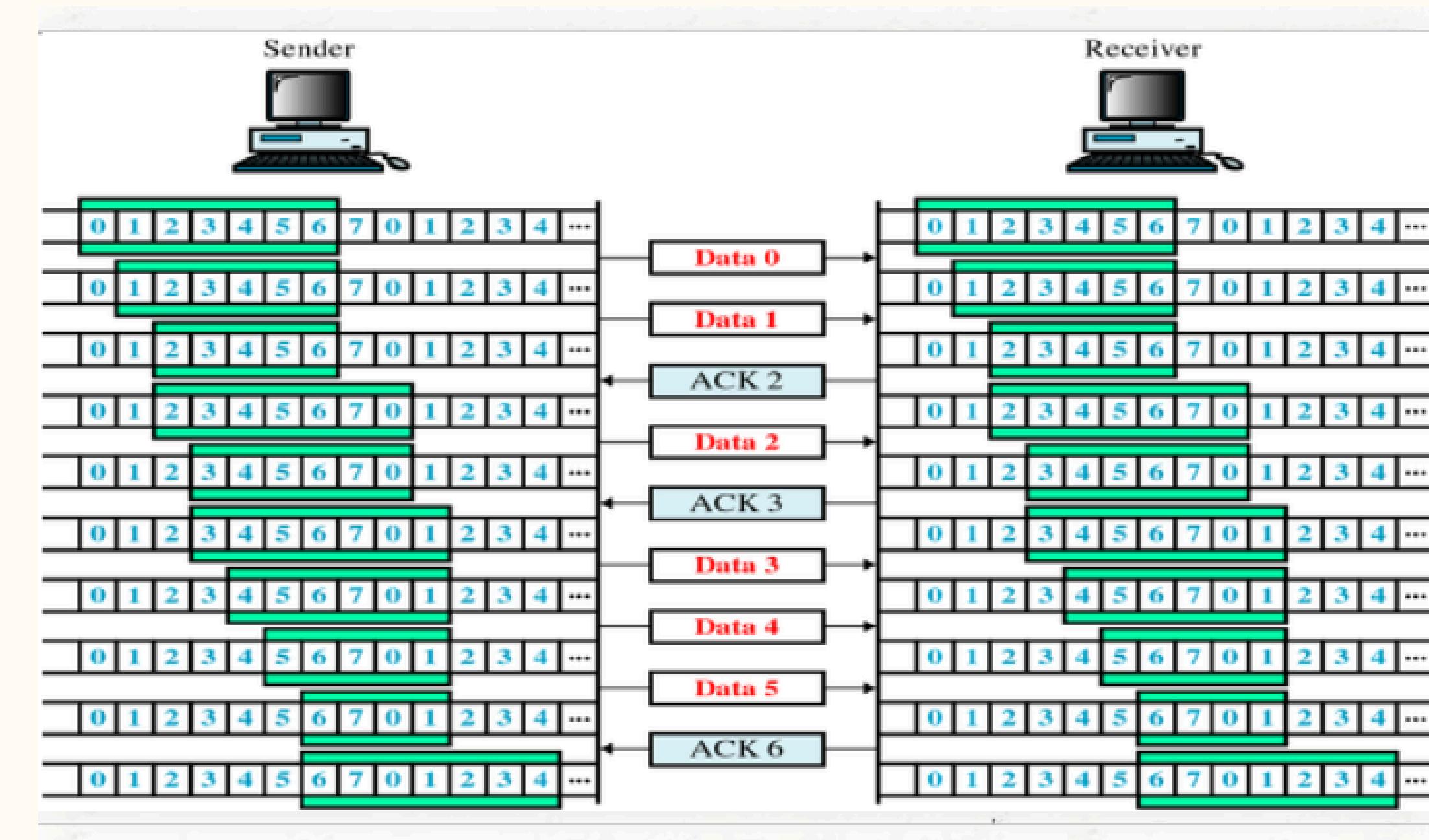
## TCP 흐름 제어 (Flow Control)

수신자: “난 지금 버퍼가 4KB 비어 있어”  
(윈도우=4KB)

송신자: 최대 4KB까지만 보냄

수신자: 일부 처리 후 “이제 2KB 더 받을 수 있어”  
(윈도우=2KB)

송신자: 그만큼만 추가 전송



## TCP 흐름 제어 (Flow Control)

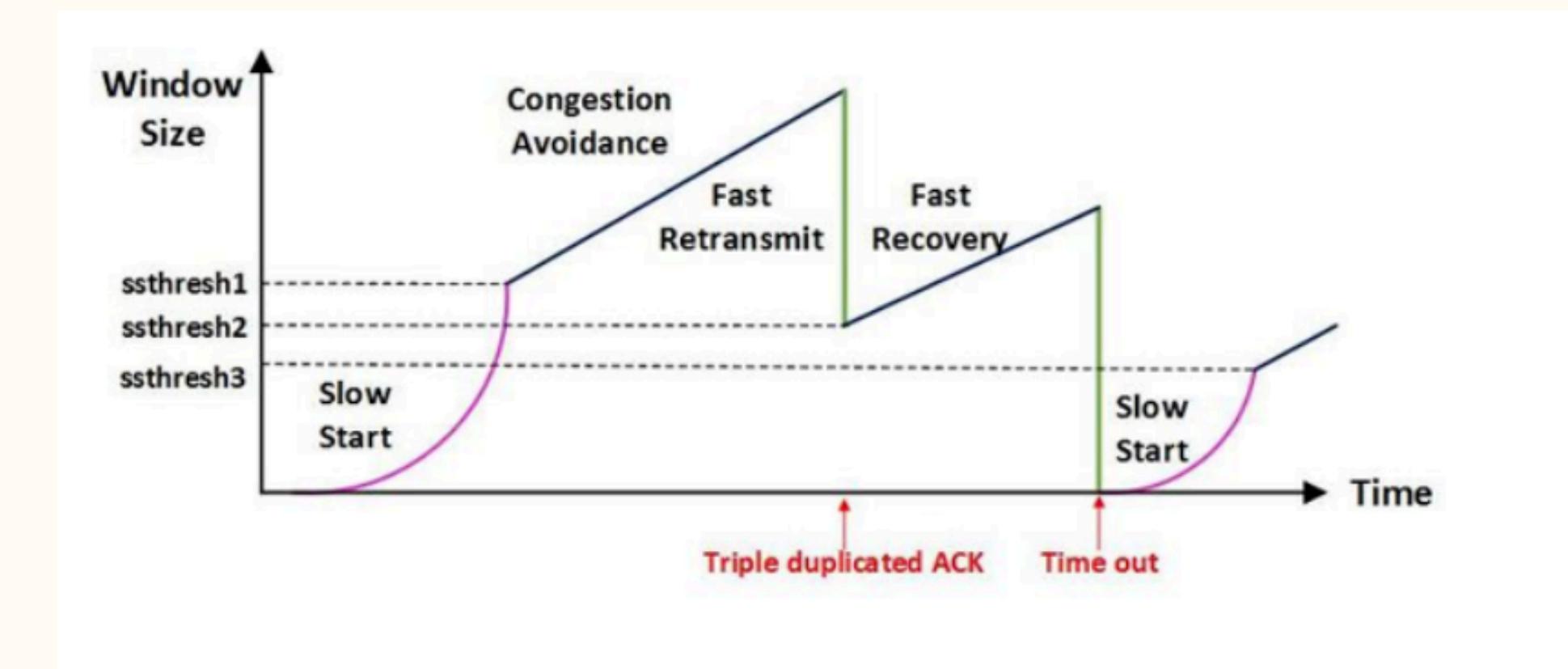
흐름 제어는 송신자 ↔ 수신자 간의 속도 불균형 문제 해결

네트워크 전체 혼잡보다는 개별 연결의 안정성에 초점

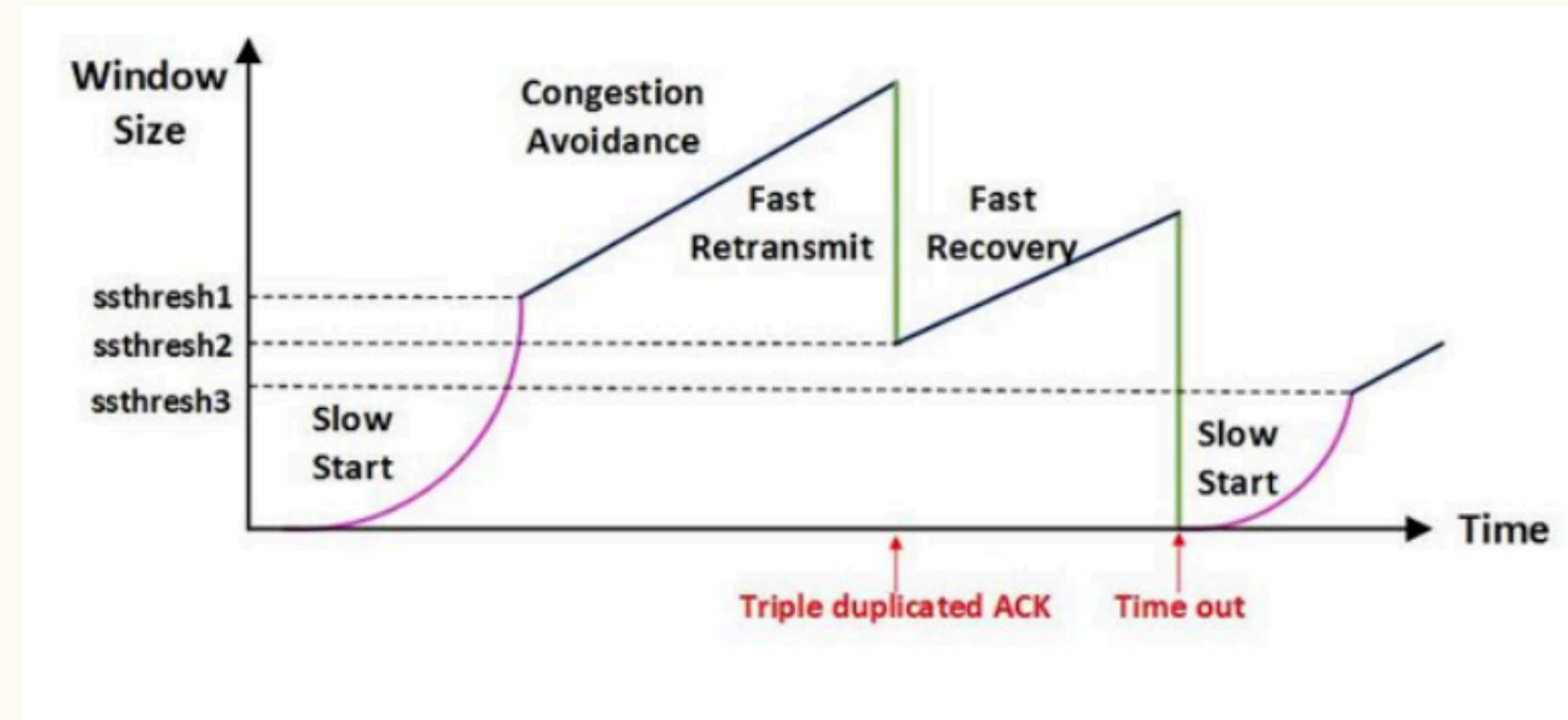
## TCP 혼잡 제어 (Congestion Control)

흐름 제어는 송수신 측 사이의 패킷 수를 제어하는 기능

혼잡 제어는 네트워크 내의 패킷 수를 조절하여 네트워크의 오버플로우를 방지하는 기능이다.



## TCP 혼잡 제어 (Congestion Control)

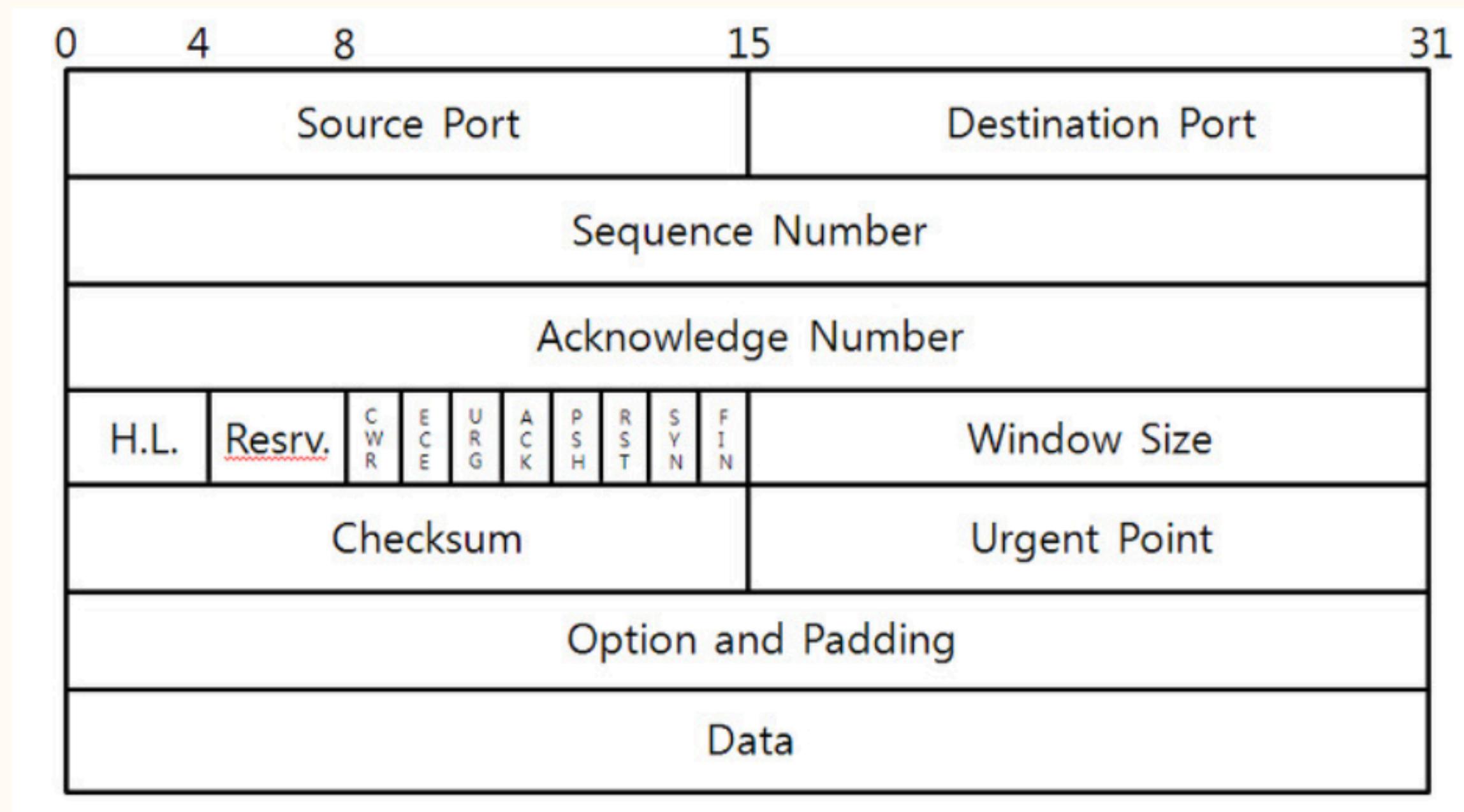


Slow Start (느린 시작)

Congestion Avoidance (혼잡 회피)

Fast Retransmit / Fast Recovery

## TCP 패킷 구조 (세그먼트)



# TCP VS UDP

PRESNTATION

프로토콜 종류	TCP	UDP
연결 방식	연결형 서비스 (패킷 교환 방식)	비연결형 서비스 (데이터그램 방식)
전송 순서	전송 순서 보장	전송 순서가 바뀔 수 있음
수신 여부 확인	수신 여부를 확인함	수신 여부를 확인하지 않음
통신 방식	1:1 통신	1:1 OR 1:N or N:N 통신
신뢰성	높다	낮다
속도	느리다	빠르다