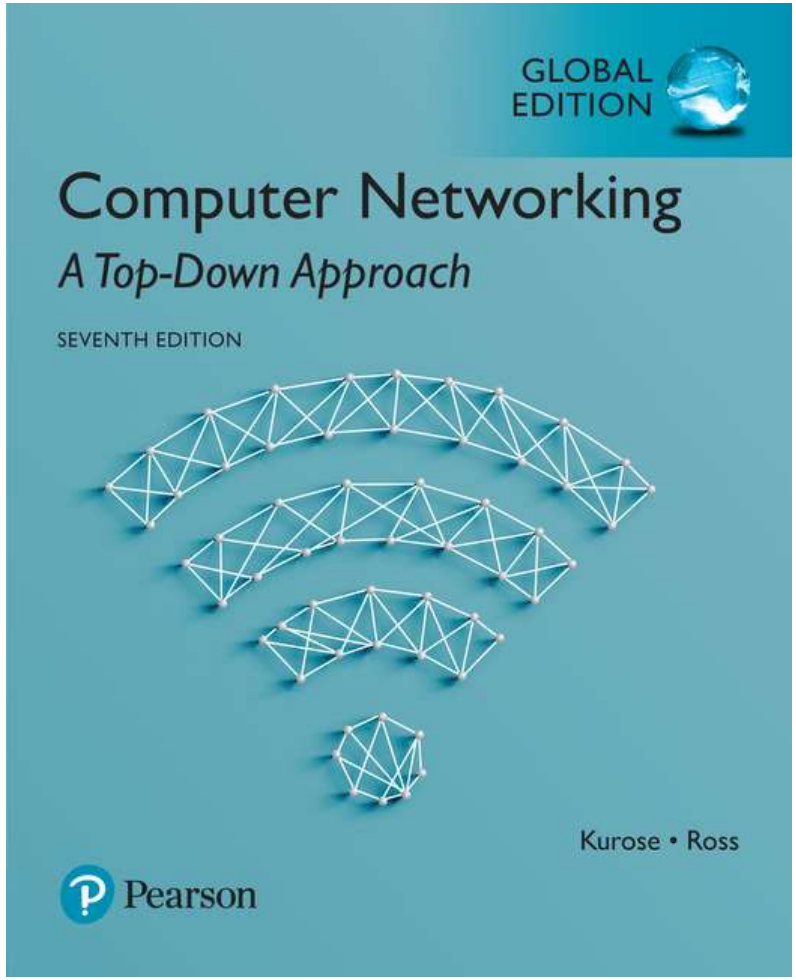
제16강
TCP 연결 관리

Computer Networking: A Top Down Approach

컴퓨터 네트워크
(2019년 1학기)

박승철교수

한국기술교육대학교
컴퓨터공학부



Chapter 3 outline

3.1 transport-layer services

3.2 multiplexing and demultiplexing

3.3 connectionless transport: UDP

3.4 principles of reliable data transfer

3.5 connection-oriented transport: TCP

- segment structure
- reliable data transfer
- flow control
- connection management

3.6 principles of congestion control

3.7 TCP congestion control

Pre-study Test :

1) TCP 연결 설정과정에서 클라이언트와 서버가 합의하지 않는 것은?

① 송신 윈도우
② 수신 윈도우
③ 포트 번호
④ 순서 번호

2) TCP에서 맨 처음 전송하는 세그먼트의 이름은?

① ACK
② URG
③ SYN
④ FIN

3) 서버에 많은 수의 TCP 연결을 한꺼번에 요청하면 어떻게 될까?

① 서버가 다운된다.
② 서버가 정상 동작한다.
③ 서버가 느려진다.
④ 서버가 서비스를 거부한다.

4) TCP의 클라이언트가 세그먼트 전송을 종료하는 FIN 세그먼트를 전송하였다.
다음 중 틀린 것은?

① 클라이언트는 데이터를 전송할 수 없다.
② 서버는 데이터를 전송할 수 없다.
③ 서버는 데이터를 수신할 수 없다.
④ 클라이언트는 정상적으로 동작한다.

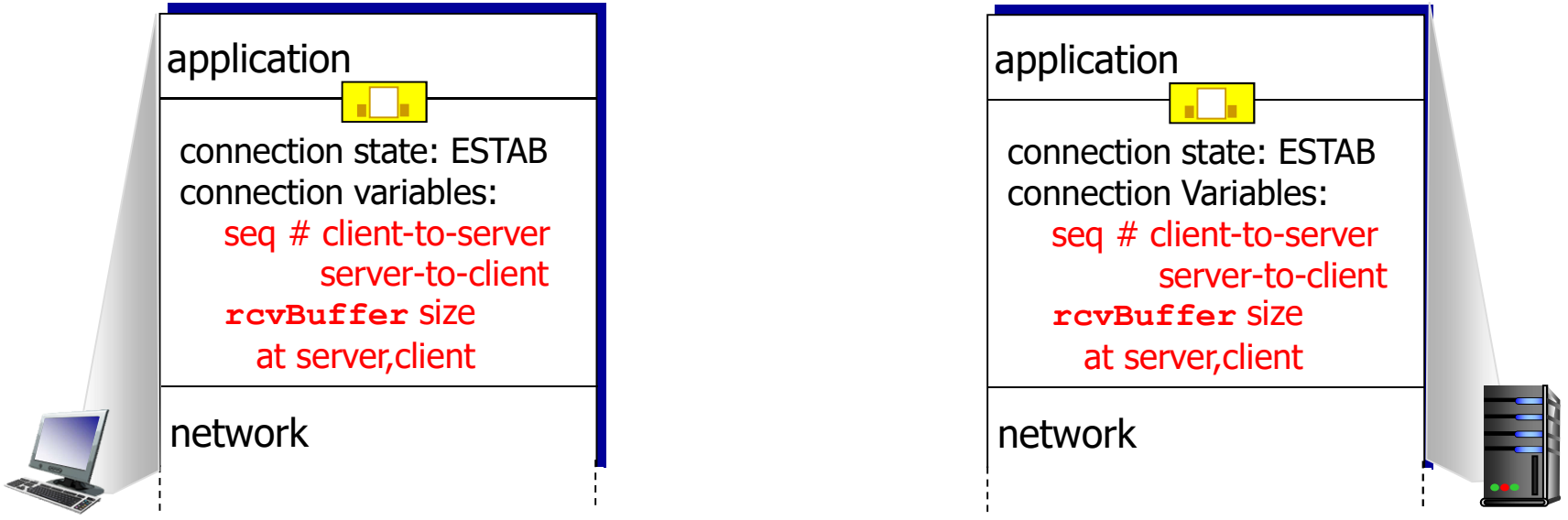
5) TCP 연결해제 과정에서 클라이언트가 서버가 송신한 FIN 세그먼트에 대한 ACK을 송신한 후 일정시간 대기하지 않고 바로 종료하면 무슨 문제가 발생할까?

① 문제가 발생하지 않는다.
② 클라이언트가 생성한 새로운 연결이 종료될 수 있다.
③ 클라이언트의 기존 연결이 종료될 수 있다.
④ 서버도 바로 종료할 수 있다.

Connection Management

before exchanging data, sender/receiver "handshake":

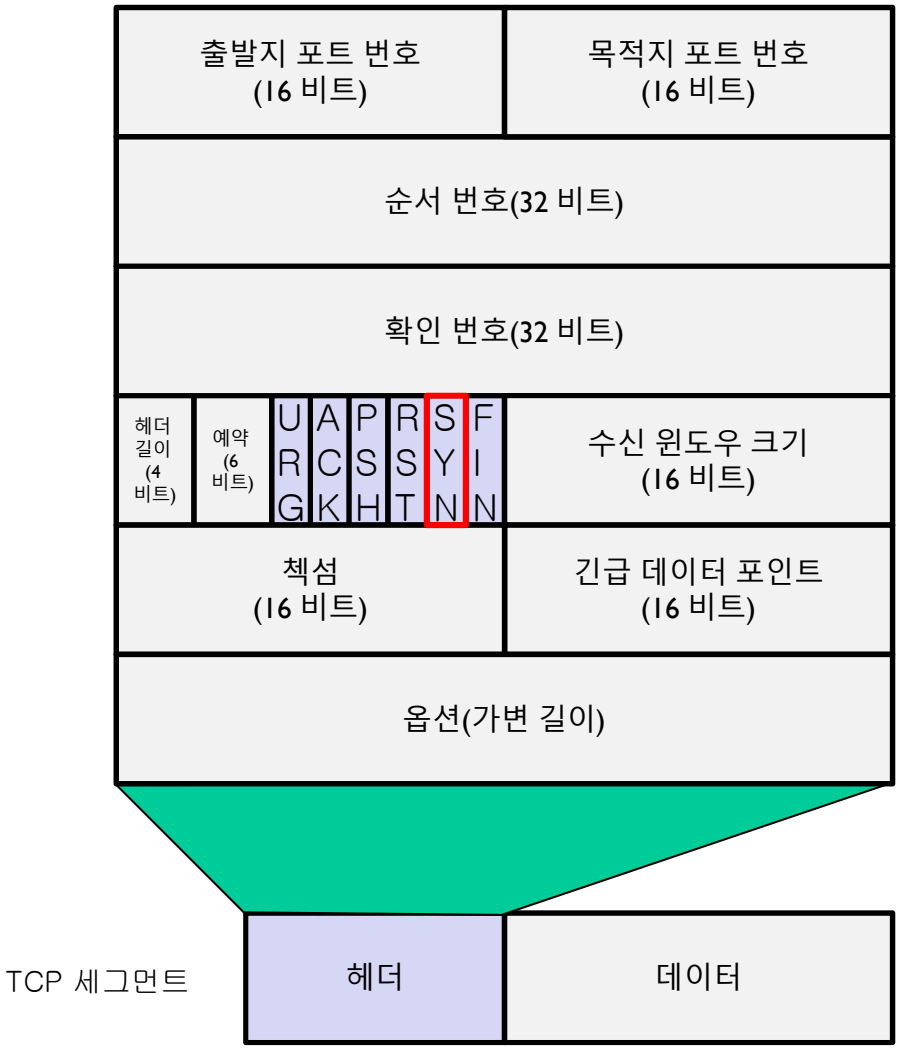
- agree to establish connection (each knowing the other willing to establish connection)
- agree on connection parameters(포트번호, 순서번호, 수신윈도우크기)



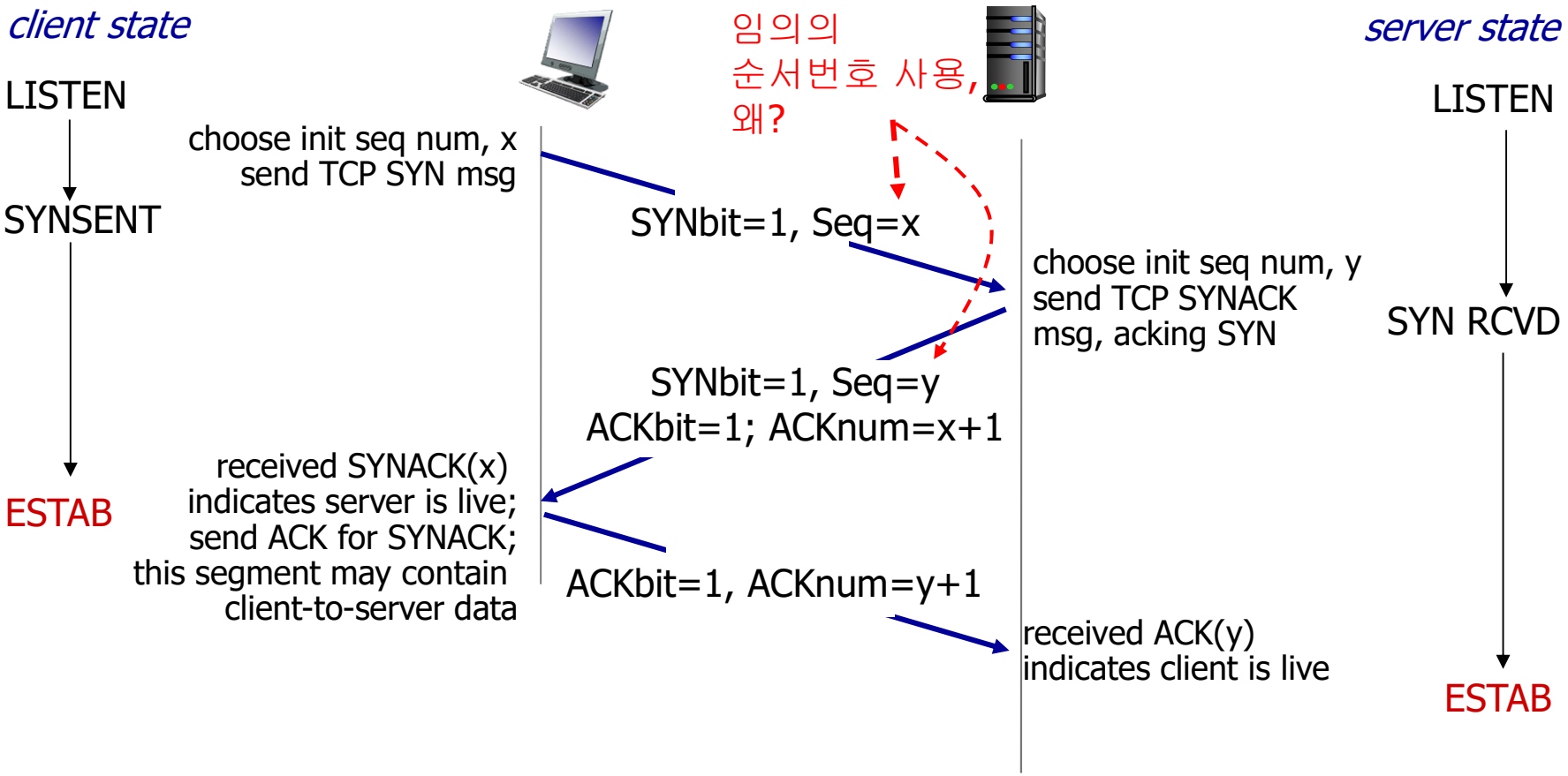

```
Socket clientSocket =
  newSocket("hostname","port
   number");
```

```
Socket connectionSocket =
   welcomeSocket.accept();
```

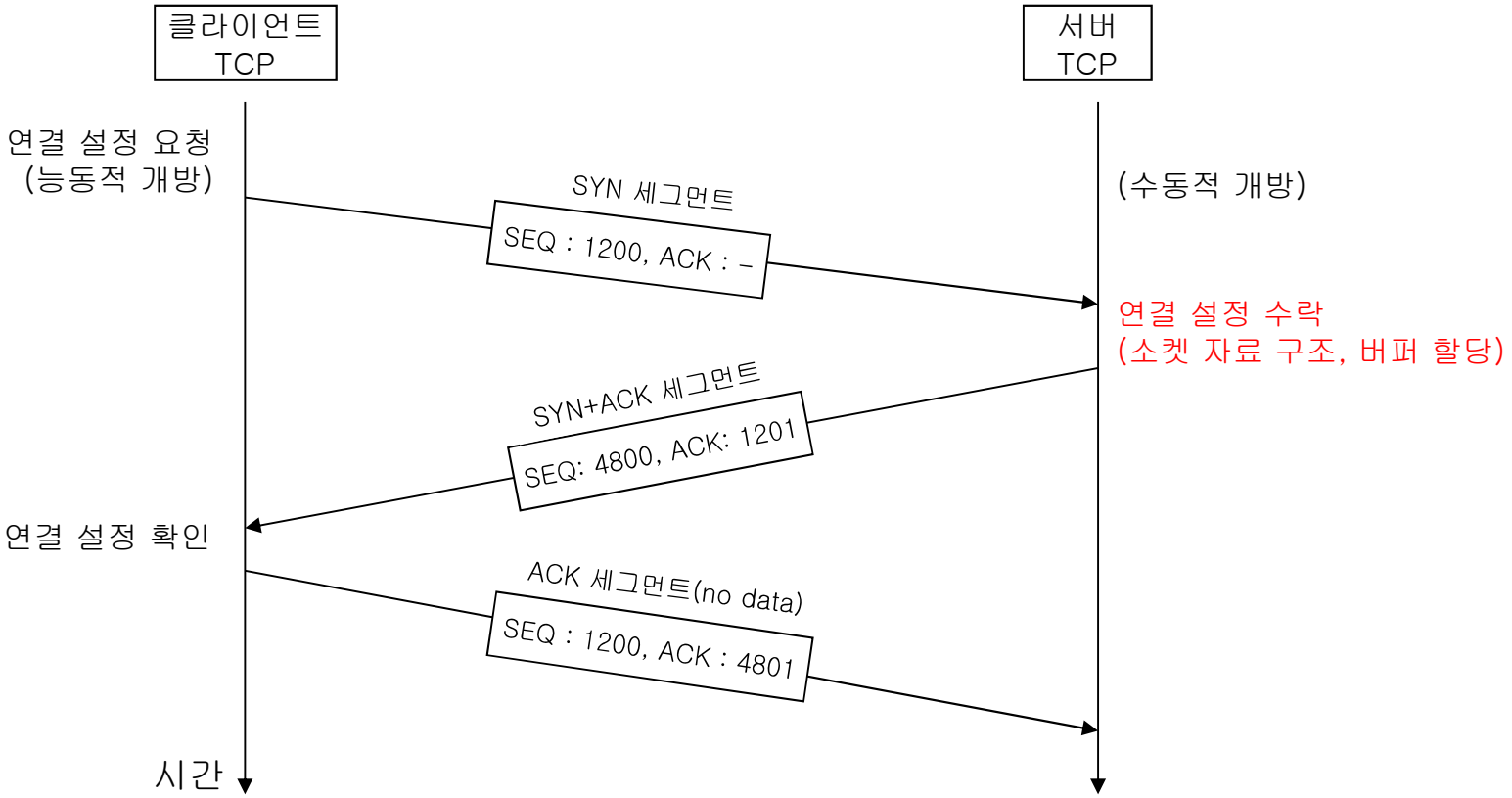
Connection Management



TCP 3-way handshake



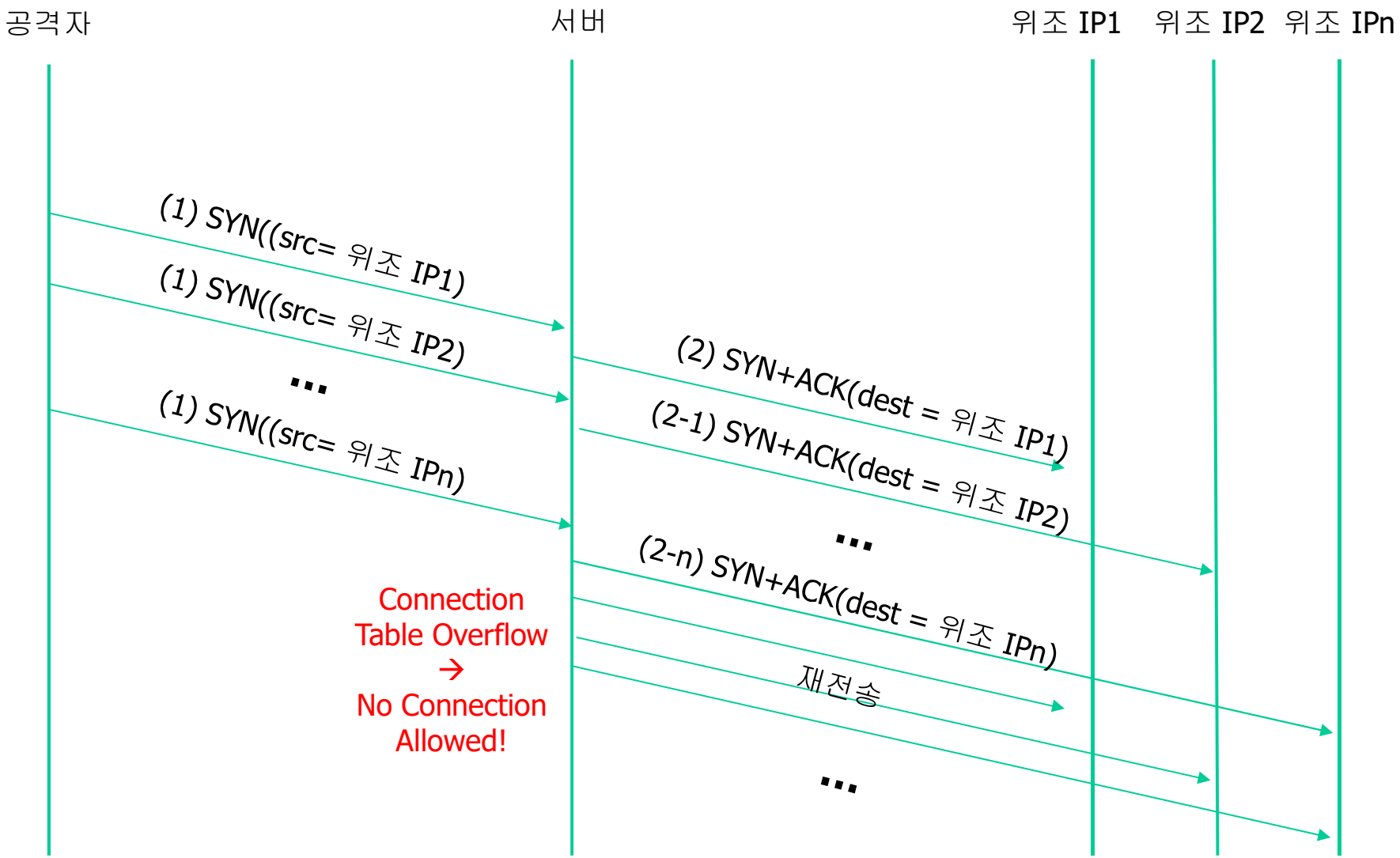
- 3-단계 핸드셰이크 기반의 연결 설정



TCP 3-way handshake

- 공격자가 특정 서버에게 의도적으로 아주 많은 수의 연결 설정을 요청하면 어떤 문제가 발생할까? – SYN Flooding 공격
- SYN Flooding 공격을 어떻게 방어할 수 있는가?

TCP SYN Flooding Attack



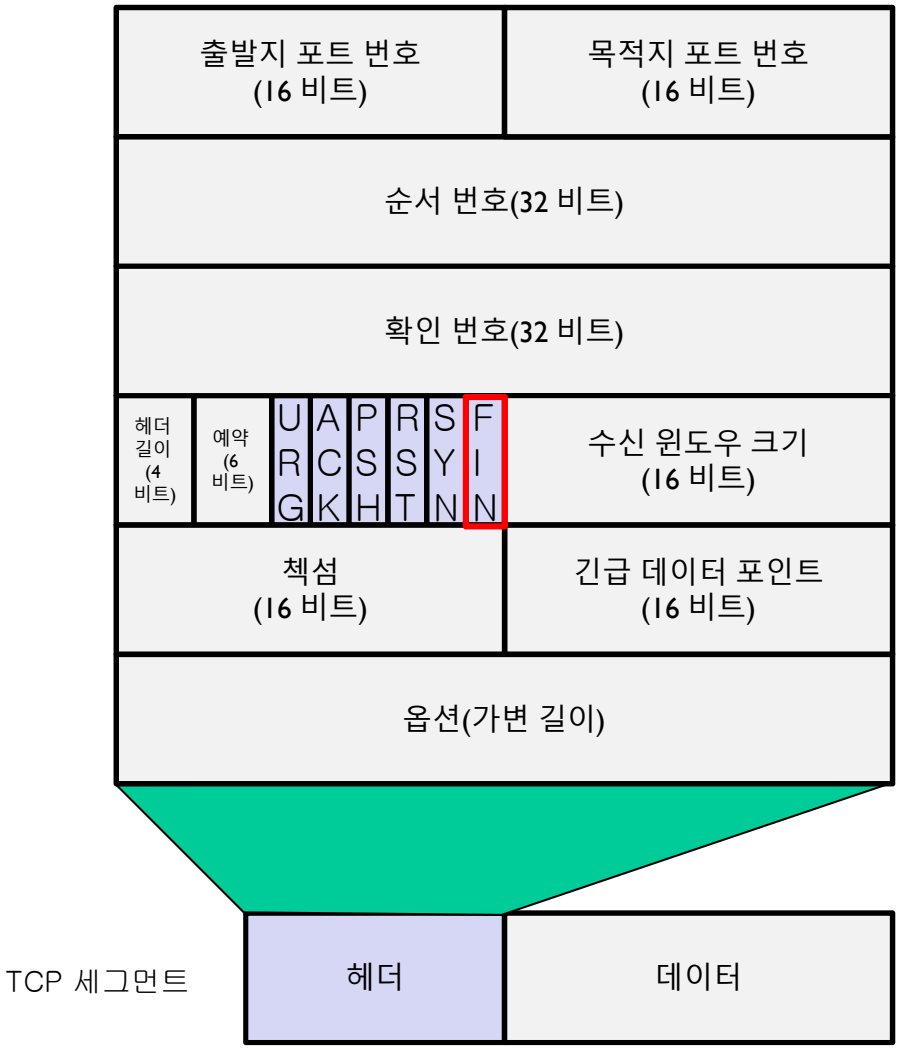
TCP SYN Flooding Attack

- SYN Cookie를 이용한 대응
  - ① SYN 패킷을 수신한 서버가 시간 정보, 클라이언트 IP 주소, 클라이언트 시작 순서 번호, 그리고 비밀 번호 등을 입력 값으로 해시값(Hash Value)를 쿠키로 구함
  - ② 쿠키를 서버 시작 순서 번호(ISN)로 하는 SYN+ACK 패킷을 클라이언트로 회신
  - ③ 정상적인 ACK 패킷을 수신할 때까지 연결 테이블에 자료구조 설정을 미룸
  - ④ 정상적인 클라이언트는 (ISN+1)을 확인 번호(ack 번호)로 하는 ACK 회신
  - ⑤ 서버가 송신한 SYN+ACK 패킷을 정상적으로 수신하지 않은 공격자는 확인번호(ack 번호)을 추론할 수 없기 때문에 ACK 패킷을 위조하여 서버에게 전송 불가

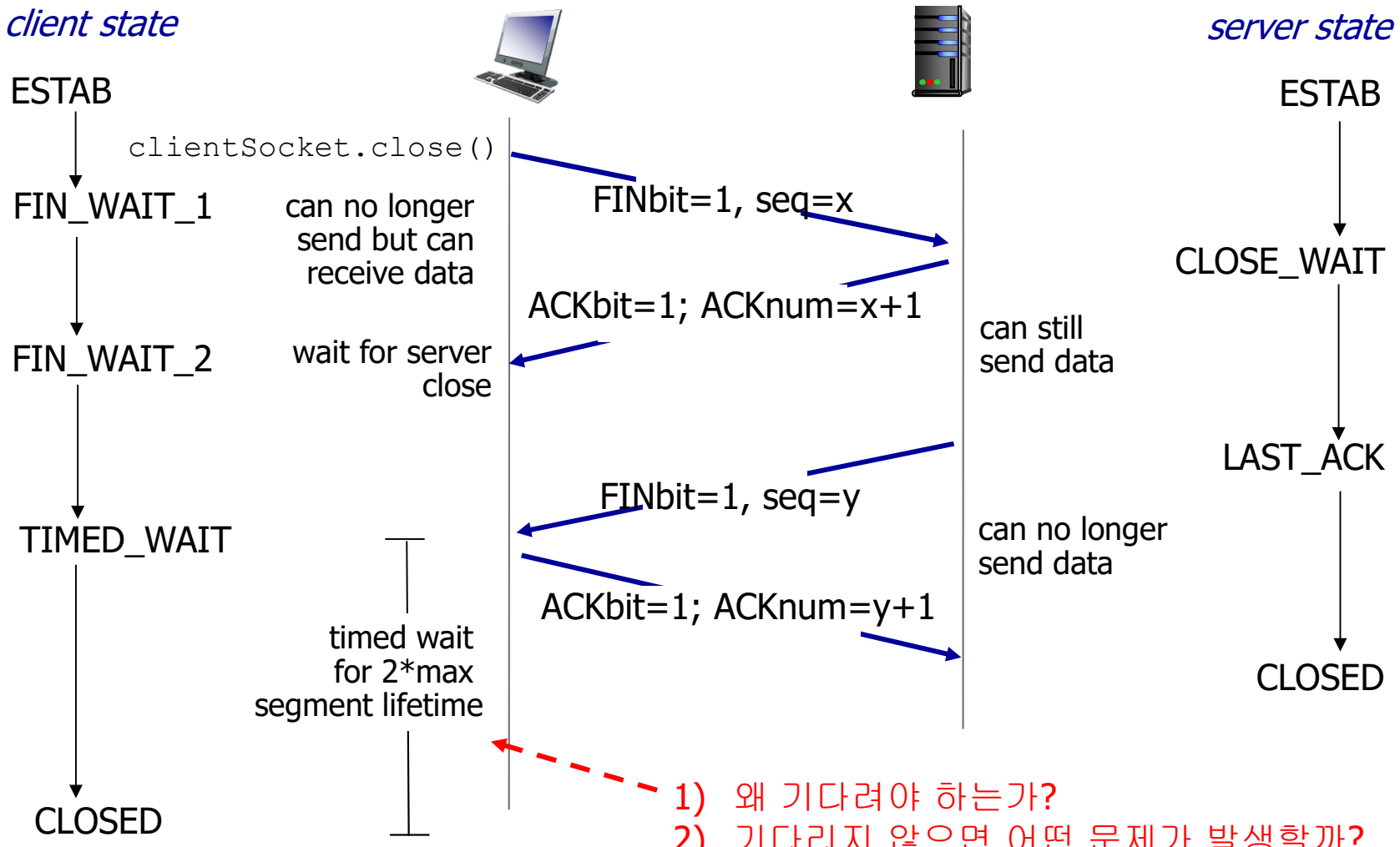
TCP: closing a connection

- client, server each close their side of connection
  - send TCP segment with FIN bit = 1
- respond to received FIN with ACK
  - on receiving FIN, ACK can be combined with own FIN
- simultaneous FIN exchanges can be handled

TCP: closing a connection



TCP: closing a connection



After-study Test :

1) TCP 연결 설정과정에서 클라이언트와 서버가 합의하지 않는 것은?

① 송신 윈도우
② 수신 윈도우
③ 포트 번호
④ 순서 번호

2) TCP에서 맨 처음 전송하는 세그먼트의 이름은?

① ACK
② URG
③ SYN
④ FIN

3) 서버에 많은 수의 TCP 연결을 한꺼번에 요청하면 어떻게 될까?

① 서버가 다운된다.
② 서버가 정상 동작한다.
③ 서버가 느려진다.
④ 서버가 서비스를 거부한다.

4) TCP의 클라이언트가 세그먼트 전송을 종료하는 FIN 세그먼트를 전송하였다.
다음 중 틀린 것은?

① 클라이언트는 데이터를 전송할 수 없다.
② 서버는 데이터를 전송할 수 없다.
③ 서버는 데이터를 수신할 수 없다.
④ 클라이언트는 정상적으로 동작한다.

5) TCP 연결해제 과정에서 클라이언트가 서버가 송신한 FIN 세그먼트에 대한 ACK을 송신한 후 일정시간 대기하지 않고 바로 종료하면 무슨 문제가 발생할까?

① 문제가 발생하지 않는다.
② 클라이언트가 생성한 새로운 연결이 종료될 수 있다.
③ 클라이언트의 기존 연결이 종료될 수 있다.
④ 서버도 바로 종료할 수 있다.