

TCP/IP 완벽 가이드

- 2-8부 TCP/IP 전송 계층 프로토콜 -

김 지 혜(jihye@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

보충

- ICMPv4 오류 메시지 유형
 - ICMPv4 목적지 접근 불가 메시지
 - 메시지 하위 유형

코드값	메시지 하위 유형	설명
0	네트워크 접근 불가	지정된 네트워크로 전달될 수 없는 경우
1	호스트 접근 불가	지정된 네트워크로 전달됐지만, 호스트에는 전달되지 않는 경우
2	프로토콜 접근 불가	목적지 호스트에서 전송 프로토콜을 사용하지 못하는 경우
3	포트 접근 불가	TCP나 UDP 헤더 속 목적지 포트가 사용되지 않는 경우
4	DF(Don't Fragment)가 켜져 있음	IPv4 라우터는 데이터그램을 자동으로 단편화하지만, DF 플래그가 켜진 데이터그램은 출발지가 단편화 원치 않음을 의미
5	소스 라우팅 실패	출발지 호스트가 경로를 결정하는 소스 라우팅 방식으로 포워딩할 수 없는 경우
6	알려지지 않은 목적지 네트워크	지정된 네트워크가 알려지지 않은 경우
7	알려지지 않은 목적지 호스트	지정된 호스트가 알려지지 않은 경우로, 보통 잘못된 주소를 의미

보충

- ICMPv4 오류 메시지 유형
 - ICMPv4 목적지 접근 불가 메시지
 - 메시지 하위 유형

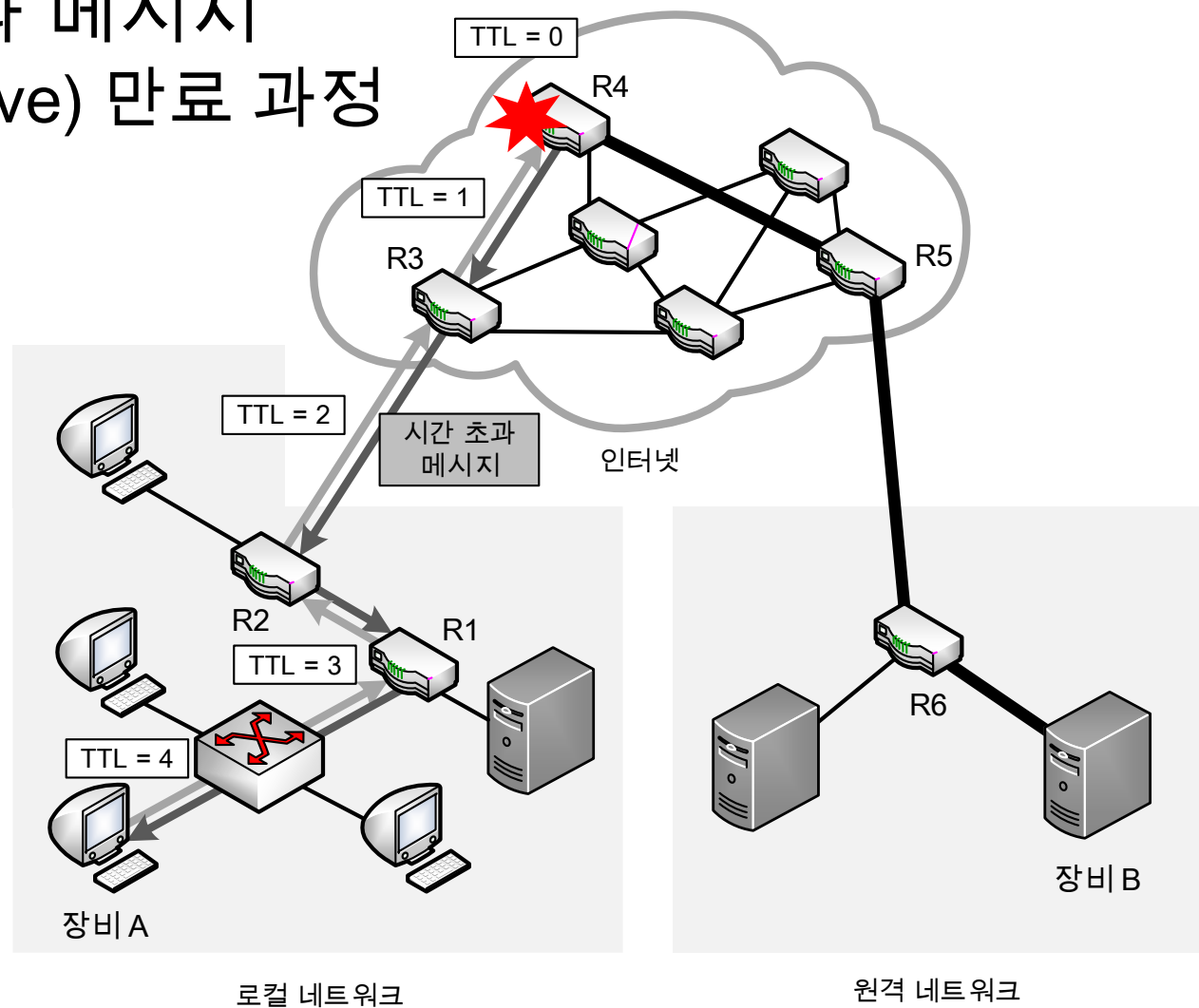
코드값	메시지 하위 유형	설명
8	출발지 호스트 고립	더 이상 쓰이지 않음
9	목적지 네트워크로의 통신이 관리상 금지	목적지 장비가 위치한 네트워크로 데이터그램을 송신하는 것이 허용되어 있지 않은 경우
10	목적지 호스트로의 통신이 관리상 금지	목적지 장비가 위치한 네트워크로 데이터그램을 송신할 수 있지만, 특정 장비로 송신할 수 없는 경우
11, 12	서비스 유형에 대한 목적지 네트워크/호스트 접근 불가	데이터그램 헤더의 서비스 유형 필드에 명시된 서비스 제공 불가, IP 주소에 지정된 목적지 네트워크/호스트에 접근 불가한 경우
13	관리상 통신 금지	데이터그램이 메시지 내용에 의해 차단되어서 전달 불가한 경우
14	호스트 우선순위 위반	서비스 유형 필드의 우선순위 값이 허용되지 않아서, 첫 번째 홉 라우터에 의해 송신되는 경우
15	우선순위 차단	받은 데이터그램의 우선순위 값이 네트워크 상 허용된 최소값보다 작아서, 라우터가 송신하는 경우

보충

- ICMPv4 오류 메시지 유형
 - ICMPv4 송신 속도 낮춤 메시지
 - 정의
 - 수신 측 버퍼 공간이 부족한 경우, 송신 측에게 송신 속도를 낮출 것을 요구하는 메시지
 - 버퍼(Buffer)란 각 장비가 데이터그램을 전달하기 위해 임시로 보관하는 장소
 - 수신 버퍼의 공간 부족
 - 송신 장비의 bps가 수신 장비의 bps보다 큰 경우
 - bps(bit per second)란 1초 동안 전송된 데이터 비트 수를 나타내는 단위
 - 송신 장비의 처리 속도가 수신 장비의 처리 속도보다 빠른 경우를 의미
 - 하나의 수신 장비가 여러 송신 장비에서 온 데이터그램을 받는 경우
 - 여러 송신 장비가 동시에 데이터그램을 보내는 경우를 포함

보충

- ICMPv4 오류 메시지 유형
 - ICMPv4 시간 초과 메시지
 - TTL(Time To Live) 만료 과정



목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

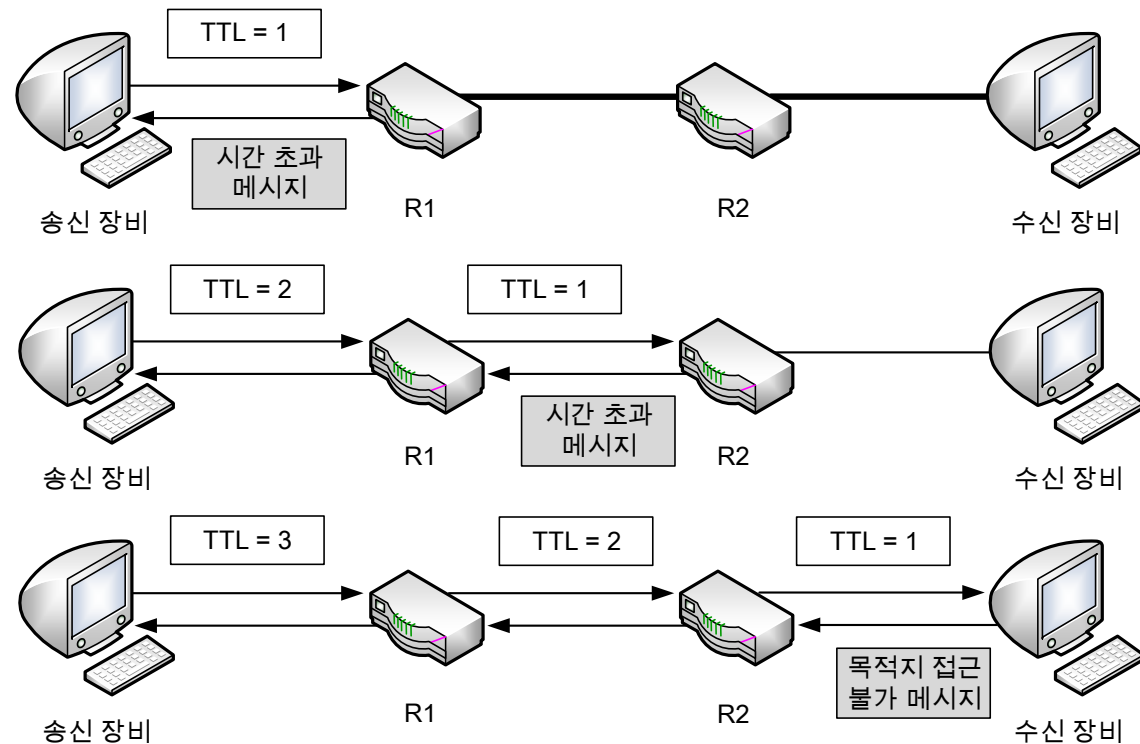
보충

- ICMPv4 정보 제공 메시지 유형
 - ICMPv4 타임스탬프 요청과 응답 메시지
 - 문제점
 - 타임스탬프 필드를 사용해도 시간 동기화가 어려움
 - 데이터그램별로 송신하는 데 걸리는 시간이 다름
 - 데이터그램을 수신하는 데 무한한 시간이 걸릴 수 있음
 - 라우터가 데이터그램을 버리기도 함
 - 시간 동기화는 어려우나, 왕복 시간 계산을 정확히 수행하기 때문에 사용
 - 해결 방안
 - 네트워크 시간 프로토콜(NTP, Network Time Protocol) 사용
 - 네트워크 상 연결된 모든 장비 간 시간 정보를 동기화하기 위한 프로토콜
 - 트리 구조로 이루어지며, 클라이언트는 최상위로부터 시간을 받아오는 것이 아니라 2,3,4단계의 서버에서 시간 정보를 가져옴

보충

- ICMPv4 정보 제공 메시지 유형
 - ICMPv4 경로 추적(Traceroute) 메시지
 - 정의
 - 목적지까지의 라우팅 경로를 추적하기 위해 쓰이는 메시지

- Traceroute 동작 원리



목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

TCP 개요

- TCP(Transmission Control Protocol)
 - 정의
 - 서버와 클라이언트 간의 신뢰성 있는 데이터 전송을 보장하는 프로토콜
 - 역사
 - 초기 TCP(Transmission Control Program)
 - 데이터그램 전송, 라우팅, 신뢰성 보장, 데이터 흐름 관리 기능 제공
 - 계층화와 모듈화라는 핵심 개념 위반
 - 기능이 각각 필요한 경우, 문제 발생
 - TCP(Transmission Control Protocol)와 IP(Internet Protocol)로 분리하기로 결정

TCP 개요

- TCP(Transmission Control Protocol)
- TCP 기능(1/2)
 - 주소 지정과 다중화
 - 포트를 이용한 주소 지정
 - 여러 소켓으로부터 데이터를 수집하여 전송하는 다중화
 - e.g., 동일한 IP 주소와 다양한 포트 번호로 메일 전송(SMTP)과 파일 전송(FTP)을 한 서버에서 동시 수행 가능
 - 연결 수립, 유지, 종료
 - 장비의 데이터 이동을 위한 연결 협상
 - 데이터 처리와 전송
 - 애플리케이션에서 데이터를 패키징하여 목적지 장비로 전달
 - 목적지 장비에서 패키징을 풀어 애플리케이션으로 전달

TCP 개요

- TCP(Transmission Control Protocol)
- TCP 기능 (2/2)
 - 신뢰성과 전송 품질 서비스 제공
 - 송신된 데이터가 순서대로 전송되어 목적지에 성공적으로 도달함
 - 흐름 및 혼잡 제어 기능 제공
 - 흐름 제어
 - 송신 측과 수신 측의 데이터 처리 속도 차이를 해결하기 위한 방식
 - 송신 측 데이터 전송량 제어
 - 혼잡 제어
 - 송신 측과 네트워크의 데이터 처리 속도 차이를 해결하기 위한 방식
 - 송신 측에서 보내는 데이터 전송 속도 제어

TCP 개요

- TCP(Transmission Control Protocol)
- TCP 한계점
 - 애플리케이션 사용 명시
 - TCP는 전송 프로토콜만 정의
 - 애플리케이션 프로토콜로 TCP 사용 방식 명시
 - 보안 제공
 - 데이터 인증이나 프라이버시를 보장하지 않음
 - 해결 방안으로 IPsec 같은 프로토콜 사용
 - 메시지 경계 유지
 - 스트림 구조이기에 메시지 경계 명시는 애플리케이션 몫
 - 실제 통신 보장
 - 승인 받지 않은 전송 탐지, 필요한 경우의 재전송 시도만 가능

TCP 개요

- TCP(Transmission Control Protocol)
- TCP 특징

특징	설명
연결형과 양방향	<ul style="list-style-type: none">• 장비 간 통신 전 연결 수립 및 양방향 송·수신
다중 연결과 종단 식별	<ul style="list-style-type: none">• 연결된 두 장비의 소켓 쌍으로 종단 식별• <클라이언트 IP 주소>:<포트 번호>, <서버 IP 주소>:<포트 번호>
신뢰성 보장	<ul style="list-style-type: none">• 모든 데이터가 목적지에 도달할 수 있도록 함• 무결성 검사를 통해 필요한 경우 재전송 처리
승인	<ul style="list-style-type: none">• 수신 여부에 대한 승인 메시지 전송
스트림 기반	<ul style="list-style-type: none">• 한 번에 한 바이트씩 연속적으로 전송되는 바이트 열• 데이터의 중복이나 손실 없이 종단 간 데이터 전송 보장

목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

TCP 원리와 일반 동작

- TCP 일반 동작의 기본 개념

- 포트(Port)

- 장비 내 여러 프로세스 중 실제로 데이터를 수신하는 프로세스 위치를 알려주는 고유한 숫자 번호

- 소켓(Socket)

- 프로세스의 데이터 송·수신을 위해 반드시 열어야 하는 창구 같은 것
- 구성 요소는 프로토콜, IP 주소, 포트 번호
 - <클라이언트 IP 주소>:<포트 번호>, <서버 IP 주소>:<포트 번호>
- TCP 동시 연결의 경우, 연결 식별을 위해 소켓 쌍 사용

TCP 원리와 일반 동작

- TCP 원리

- TCP 데이터 처리 방식

- 애플리케이션에서 송·수신 시 TCP는 세그먼트를 바이트 스트림 처리하는 스트림 동작 방식 사용

- TCP 데이터 패키징 단위

- 수신한 스트림을 IP를 위해 분리한 세그먼트 사용
- 세그먼트 크기는 최대 세그먼트 크기(MSS, Maximum Segment Size)로 결정

- TCP 데이터 식별 방식

- 데이터가 순서대로 목적지에 도달했는지 순서 번호를 통해 확인
 - 수신한 세그먼트는 순서 번호를 이용하여 원본 데이터 스트림으로 재조합
 - 손실된 데이터 확인 시 재전송하는 경우에 사용

TCP 원리와 일반 동작

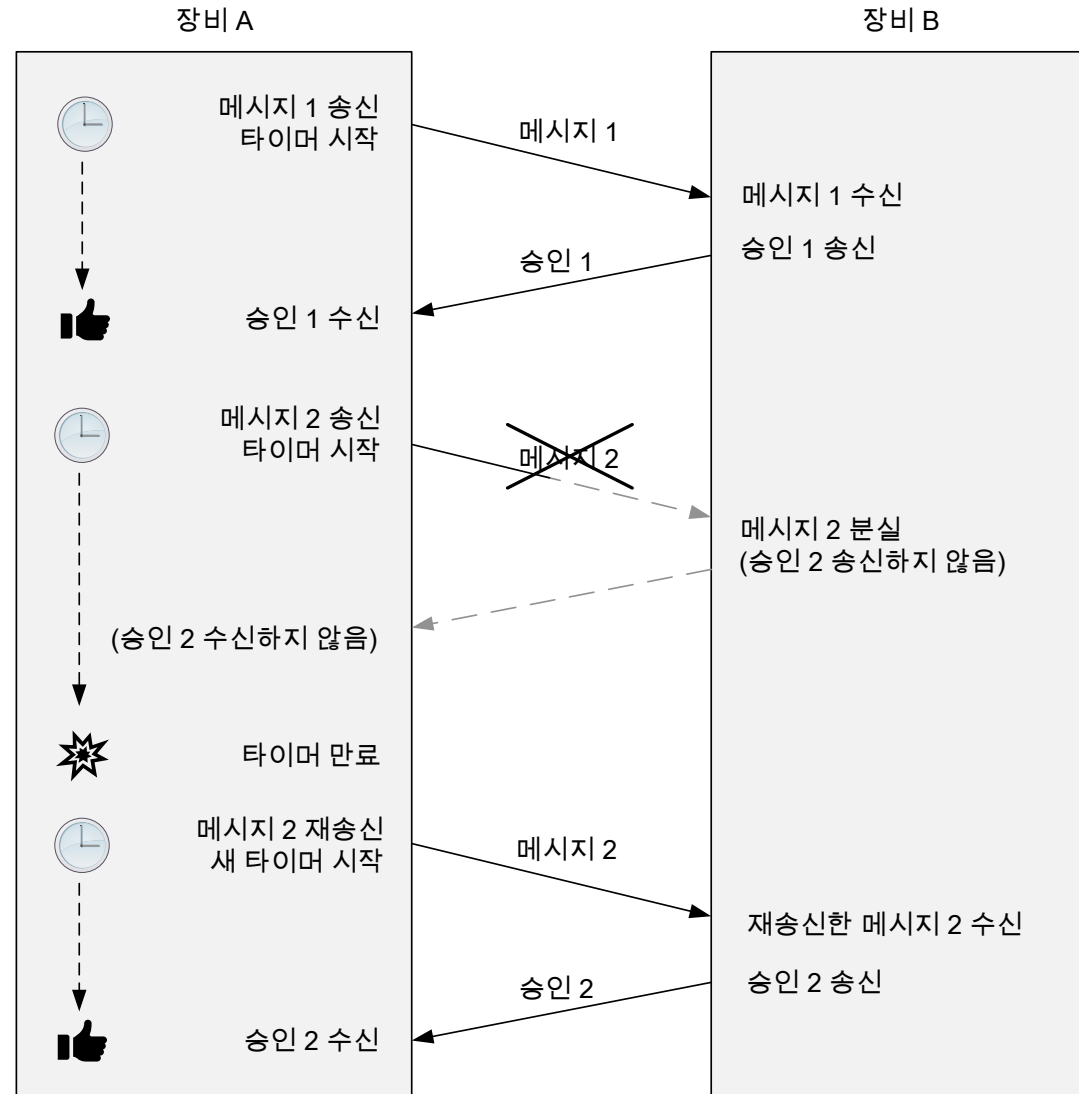
- TCP 슬라이딩 윈도우 승인 체계
 - 두 개의 장비 간 세그먼트 흐름 제어를 위한 방법
 - 재전송을 사용하는 긍정 승인(PAR, Positive Acknowledgment with Retransmission) 사용
 - 사용하는 이유
 - 신뢰성 보장
 - 송신 데이터의 목적지 도달 여부 확인
 - 도달하지 못한 경우 데이터 재전송
 - 데이터 흐름 제어
 - 송신 장비의 데이터 처리 속도가 수신 장비의 데이터 처리 속도보다 빠른 경우
 - 수신 장비가 처리 가능한 속도로 송신 장비의 데이터 송신을 관리

TCP 원리와 일반 동작

- TCP 슬라이딩 윈도우 승인 체계

- 긍정 승인(PAR) 동작 방식

- 송신 메시지에 대한 승인
받기 전까지 다음 메시지
전송 불가



TCP 원리와 일반 동작

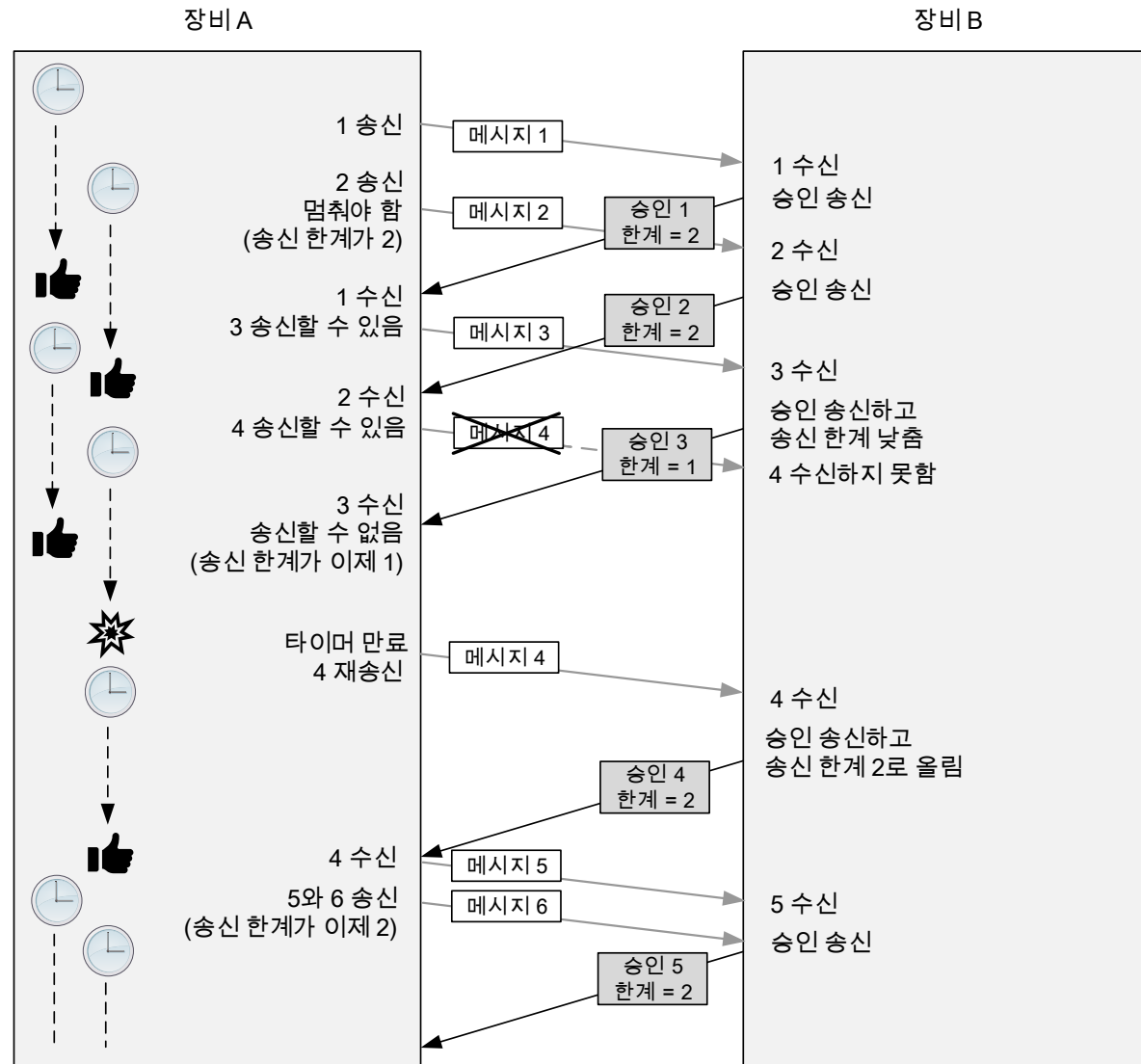
• TCP 슬라이딩 윈도우 승인 체계

• 개선된 PAR

- 송신 장비가 여러 메시지 동시에 송신 가능

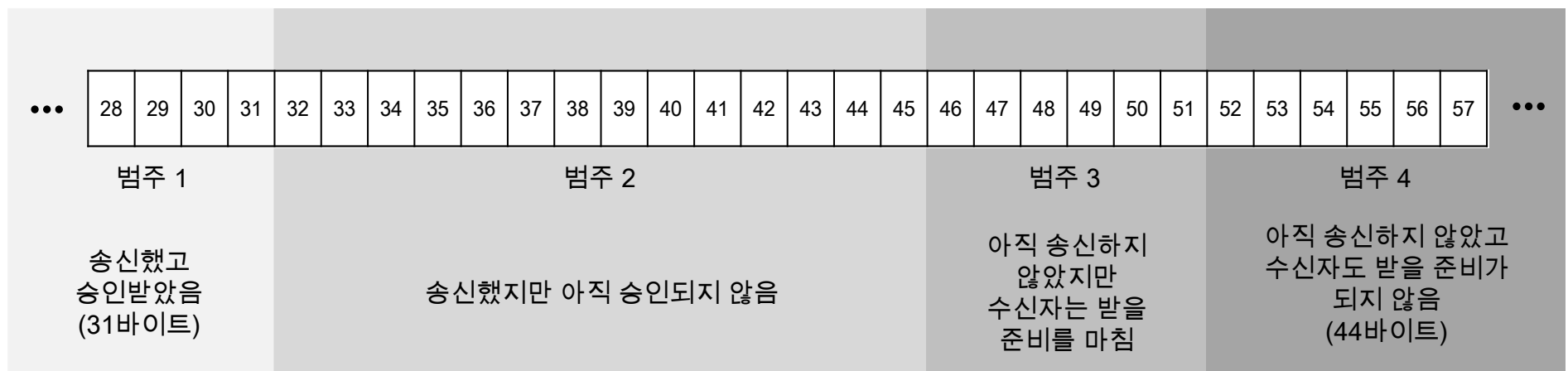
• 개선된 사항

- 순서 번호
 - 개별적으로 승인 가능
 - 동시에 여러 메시지 수신 가능
- 송신 제한 필드
 - 송신 가능한 메시지의 최대 수 지정 가능
 - 메시지 송신 속도를 제한함으로써 흐름 제어



TCP 원리와 일반 동작

- TCP 슬라이딩 윈도우 승인 체계
 - 윈도우 개념
 - 윈도우(Window)
 - 메모리 버퍼의 일정 영역
 - 슬라이딩 윈도우(Sliding Window)
 - 수신 측에서 설정한 윈도우 크기만큼 송신 측에서 세그먼트를 전송할 수 있게 하여 데이터 흐름을 제어하는 방법
- TCP 전송 스트림 바이트 상태



TCP 원리와 일반 동작

- TCP 슬라이딩 윈도우 승인 체계

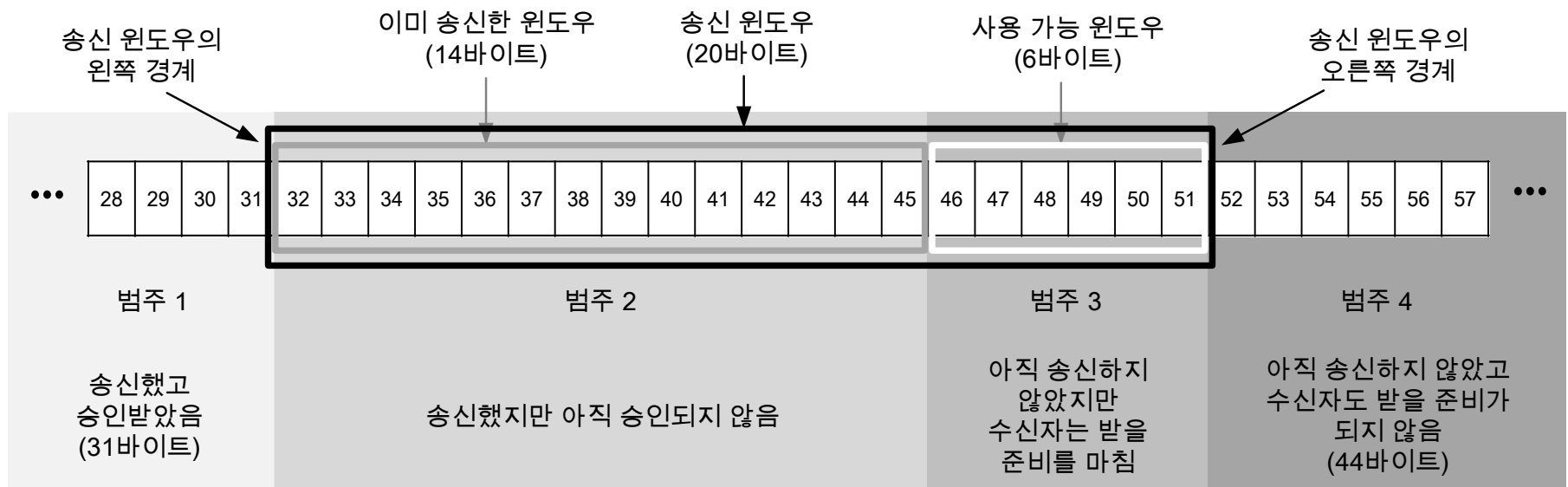
- 윈도우 크기

- 송신 윈도우(Send Window)

- 송신장비가 승인 받기 전, 한 번에 송신 가능한 최대 데이터 바이트 수

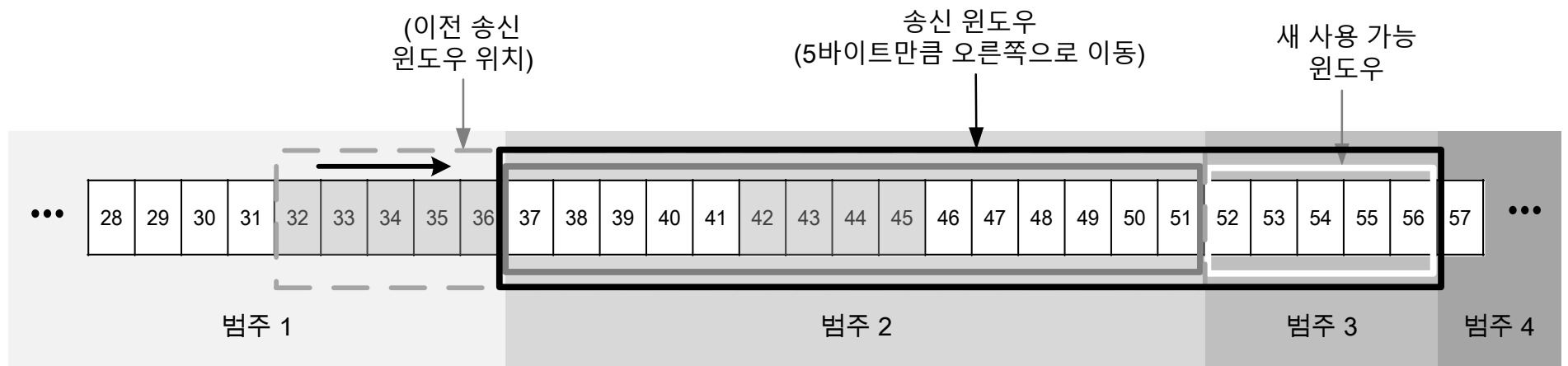
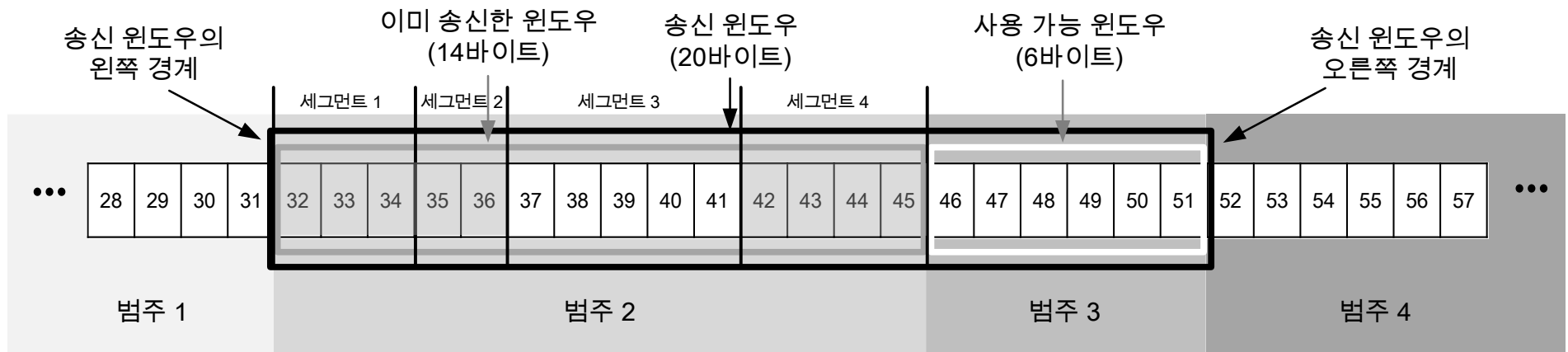
- 사용 가능 윈도우(Usable Window)

- 송신장비가 현 시점에서 송신 가능한 최대 데이터 바이트 수
 - (사용 가능 윈도우) = (송신 윈도우) - (이미 송신한 윈도우)



TCP 원리와 일반 동작

- TCP 슬라이딩 윈도우 승인 체계
- 송신 윈도우 슬라이딩과 빠진 승인 처리



목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

TCP 기본 동작

- TCP 동작 방식: 유한 상태 머신(FSM, Finite State Machine)
 - 정의
 - 유한한 상태를 가지는 머신
 - 특징
 - 연결 수립, 관리, 종료 과정을 표현
 - 주로 한 장비가 연결 수립이나 종료를 시작, 다른 장비가 응답
- 기본 FSM 개념

기본 FSM 개념	설명
상태	특정 시간에 프로토콜 소프트웨어가 처한 상황
전이	한 상태에서 다른 상태로 움직이는 행위
이벤트	상태 간 전이를 하게 만든 어떤 일
행동	장비가 이벤트에 대한 반응으로 다른 상태로 전이하기 전에 하는 일

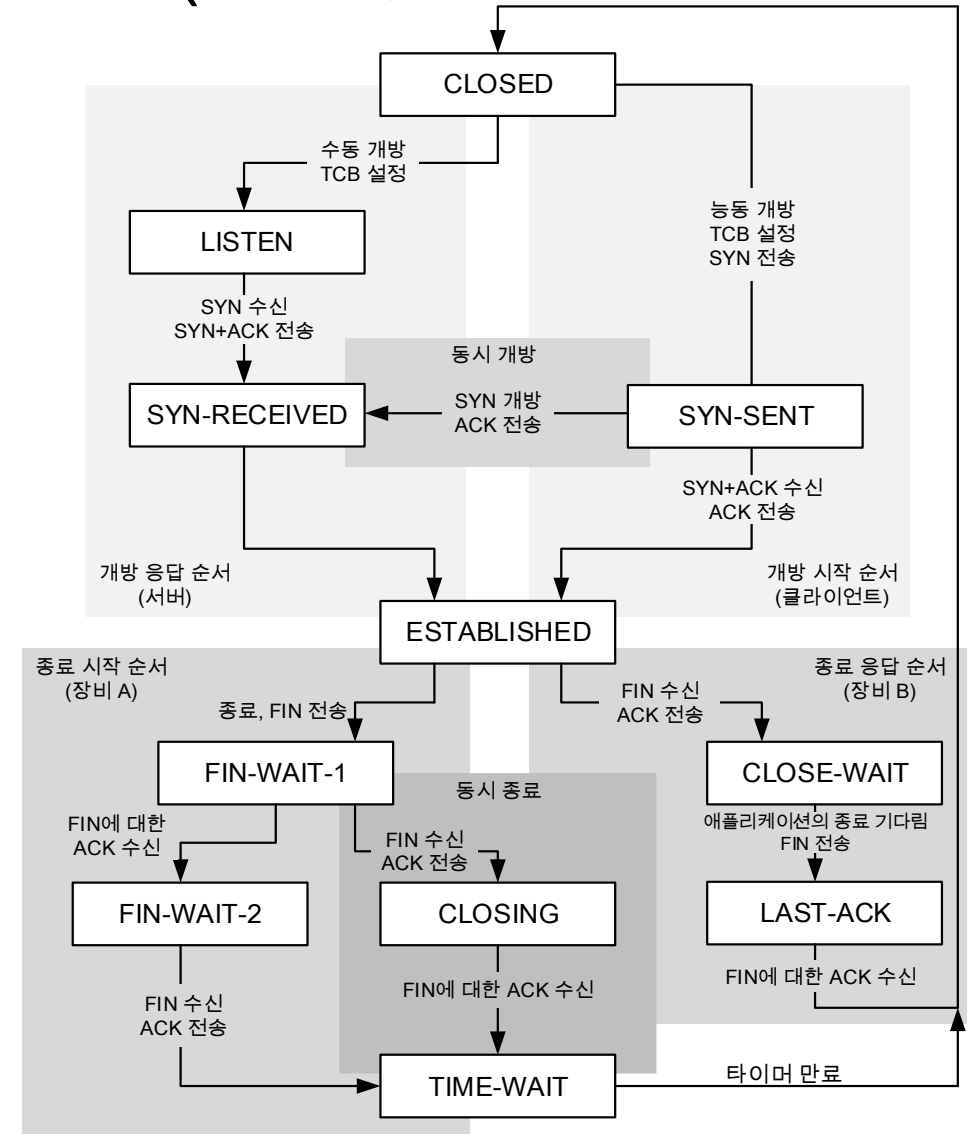
TCP 기본 동작

- TCP 동작 방식: 유한 상태 머신(FSM, Finite State Machine)
- FSM 주요 메시지
 - SYN(Synchronize) 메시지
 - 연결 초기화 및 수립
 - 장비 간 순서 번호 동기화시킴
 - FIN(Finish) 메시지
 - 연결 종료하고 싶은 상태임을 알림
 - ACK(Acknowledgment) 메시지
 - SYN 메시지 혹은 FIN 메시지를 수신한 상태임을 알림

TCP 기본 동작

• TCP 동작 방식: 유한 상태 머신(FSM, Finite State Machine)

상태	설명
CLOSED	연결 없음 상태 (수립 전 상태 혹은 연결 종료 상태)
SYN-SENT	클라이언트 SYN 송신 후 대기 상태
LISTEN	클라이언트 SYN 수신을 위한 대기 상태
SYN-RECEIVED	클라이언트 ACK+서버 SYN 송신 후 대기 상태
ESTABLISHED	연결 수립 상태
FIN-WAIT-1	장비 A 혹은 두 장비의 FIN 송신 후 대기 상태
CLOSE-WAIT	장비 A ACK 송신 후 대기 상태
FIN-WAIT-2	장비 A ACK 수신 후 장비 B FIN 수신 대기 상태
LAST-ACK	장비 B FIN 송신 후 대기 상태
CLOSING	두 장비가 서로 ACK 송신 후 대기 상태
TIME-WAIT	장비 B 혹은 두 장비가 서로 ACK 수신 후 최종 종료를 위한 대기 상태



TCP 기본 동작

- TCP 연결 준비
 - 전송 제어 블록(TCB, Transmission Control Block)
 - 연결을 식별하기 위한 데이터 구조
 - 연결 정보 저장
 - 연결 식별을 위한 두 소켓 번호
 - 송·수신 데이터를 가진 버퍼를 가리키는 포인터
 - 승인하거나 승인하지 않은 바이트 수, 현재 윈도우 크기 등을 추적하는 변수
- 클라이언트와 서버의 개방 동작
 - 능동 개방
 - 클라이언트는 서버에게 SYN 메시지를 보내 연결 시작
 - 수동 개방
 - 서버는 클라이언트의 연결 요청을 받아들일 준비 동작

TCP 기본 동작

• TCP 연결 수립

- 두 장비가 메시지를 교환하여 초기 접속 상태(CLOSED)에서 정상 동작 상태(ESTABLISHED)로 전이함
- 연결 수립 기능(1/2)
 - 접촉과 통신
 - 클라이언트와 서버는 서로 접촉하여 메시지 전송 후 통신 시작
 - 접촉과 통신 과정을 위한 제어 메시지
 - 연결 초기화에 사용하는 세그먼트임을 알리는 SYN 메시지
 - 세그먼트 전송 장비에게 메시지를 수신했음을 알리는 ACK 메시지

• 인자 교환

인자 교환 방법	설명
윈도우 크기	16비트인 TCP 윈도우 크기보다 더 큰 값 사용 가능
선택적 승인 허용	잃어버린 세그먼트만을 재전송할 수 있도록 선택적 승인 옵션 사용 가능
대체 체크섬 방식	표준 TCP 체크섬이 아닌 다른 체크섬 방법 사용 가능

TCP 기본 동작

- TCP 연결 수립

- 연결 수립 기능(2/2)

- 순서 번호 동기화

- 순서 번호는 전송하는 데이터의 순서
 - 승인 번호는 다음에 전송해야 하는 데이터의 시작 위치
 - 상대 장비가 보낸 순서 번호 + 1(또는 수신한 데이터 바이트)

- 동작 과정

1. 클라이언트의 연결 요청

- 순서 번호 필드에 클라이언트의 초기 순서 번호(ISN, Initial Sequence Number)를 넣어 SYN 전송

2. 서버의 승인과 연결 요청

- 승인 번호 필드에 클라이언트 순서 번호+1 값을 넣고,
순서 번호 필드에 서버의 ISN을 넣어 SYN+ACK 전송

3. 클라이언트의 승인

- 승인 번호 필드에 서버 순서 번호+1 값을 넣어 ACK 전송

TCP 기본 동작

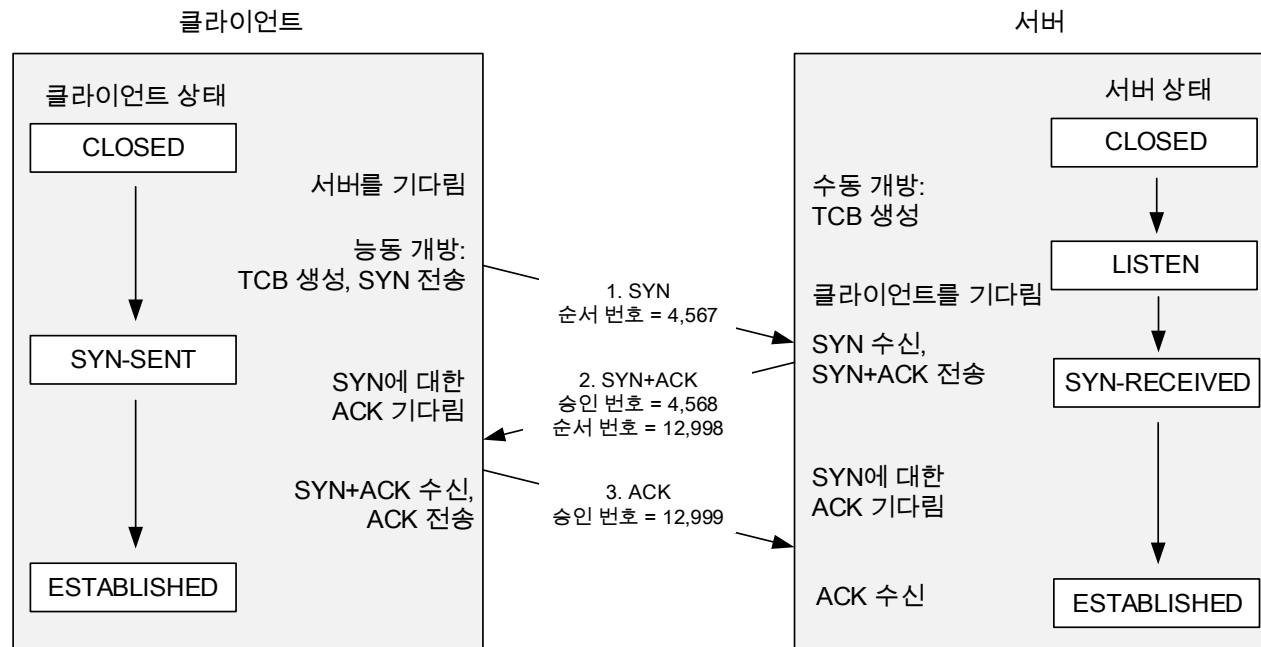
- TCP 연결 수립

- 연결 수립 방법(1/2)

- 쓰리 웨이 핸드셰이크(Three-way Hand-shake)

- 한 장비가 닫힌 상태에서 연결하는 메시지 교환 과정

1. 클라이언트가 SYN 전송
2. 서버는 1에 대한 ACK와 자신의 SYN를 합쳐서 전송
3. 클라이언트는 서버의 SYN에 대한 ACK 전송



TCP 기본 동작

- TCP 연결 수립

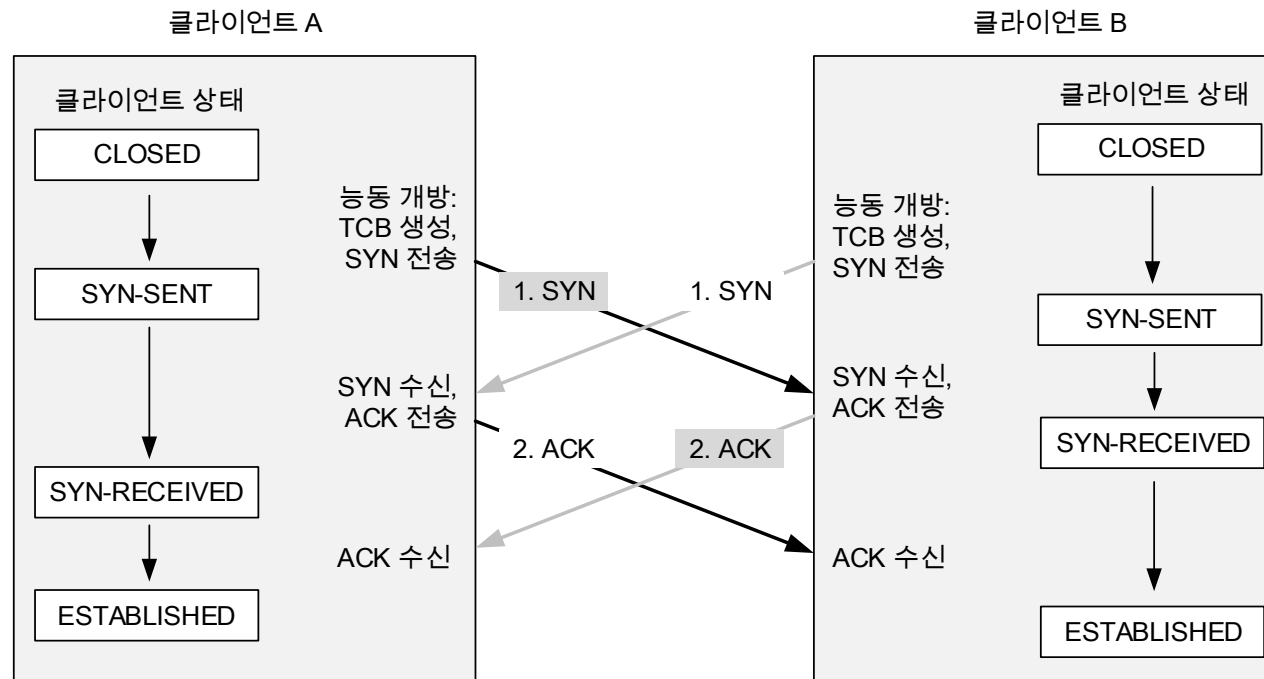
- 연결 수립 방법(2/2)

- 동시 개방

- 두 장비가 상대 장비로 동시에 연결하는 메시지 교환 과정

- 쓰리 웨이 핸드셰이크 방식과의 차이

- 동시 개방은 SYN을 받고 ACK를 보내고, 쓰리웨이는 SYN와 ACK를 함께 보냄



TCP 기본 동작

- TCP 연결 관리 문제

- TCP 초기화

- 초기화 세그먼트의 순서 번호가 유효한지 검사하는 것
 - 헤더의 RST(Reset) 플래그가 1로 설정된 세그먼트

- 연결을 초기화해야 하는 상황

- 세그먼트를 보낸 장비와 연결을 맺고 있지 않는 경우
- 연결을 기다리는 프로세스가 없는 포트에 SYN을 수신한 경우
- 부정확한 순서/승인 번호를 가지는 메시지를 수신한 경우
 - 메시지가 이전 연결에 속해있거나 조작된 것을 의미
- 상대 장비에게 알리지 않은 채 연결을 닫거나 중지하는 경우
 - 반 개방 연결(Half-open Connection)
 - 한 장비는 ESTABLISHED 상태, 다른 장비는 CLOSED 상태

TCP 기본 동작

- TCP 연결 관리 문제

- TCP 초기화

- TCP 초기화 세그먼트 처리 방식

- LISTEN 상태로 초기화하는 경우

- 장비가 LISTEN 상태인 경우

- 초기화 세그먼트 무시하고 그 상태에 머무름

- 장비가 SYN-RECEIVED 상태 이전에 LISTEN 상태에 있던 경우

- LISTEN 상태로 돌아감

- 이유

- 순서 번호가 제대로 전달된 상태가 아니기에, 순서 번호를 다시 설정하고 전송할 수 있는 상태이기 때문

- CLOSED 상태로 초기화하는 경우

- 모든 나머지 경우

- 연결을 종료하고 CLOSED 상태로 돌아감

- 이유

- 순서 번호가 다 전달된 후이기에, 처음부터 시작하는 것이 효율적이기 때문

TCP 기본 동작

- TCP 연결 관리 문제
 - 유힤(Idle) 연결 처리
 - 유힤 연결
 - 연결이 오랜 기간 동안 정지 상태에 있는 것
 - 유힤 연결 처리 방식
 - 킵얼라이브(Keep-alive) 메시지
 - 유힤 연결을 처리하는 메시지
 - 다른 장비에게 연결이 수립된 상태임을 알려줌
 - 연결이 유효한 경우, 상대 장비는 승인을 포함하는 세그먼트 전송
 - 연결이 유효하지 않은 경우, 상대 장비는 초기화 세그먼트 전송

TCP 기본 동작

- TCP 연결 종료

- 두 장비 모두 데이터를 보내지 않는 상태

- TIME-WAIT 상태 필요한 이유

- 일반적인 연결 종료 과정

- 상대 장비의 ACK 수신을 확신하기 위한 대기 시간 필요

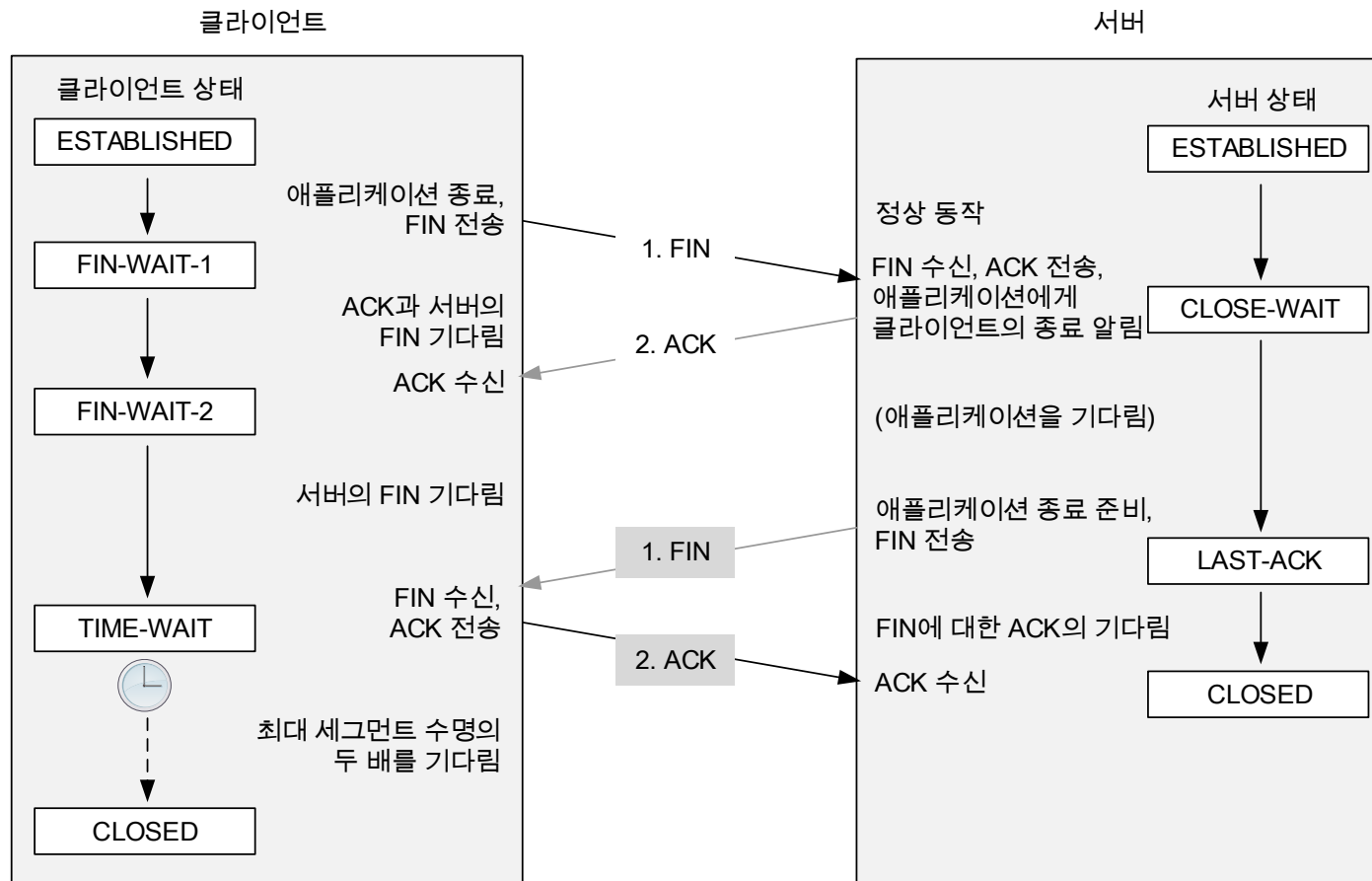
- 동시 연결 종료 과정

- 한 연결의 종료와 새로운 연결 사이의 일정 시간 필요

- 일정 시간이란 보통 최대 세그먼트 수명(MSL, Maximum Segment Lifetime)의 2배인 120*2초로 정의

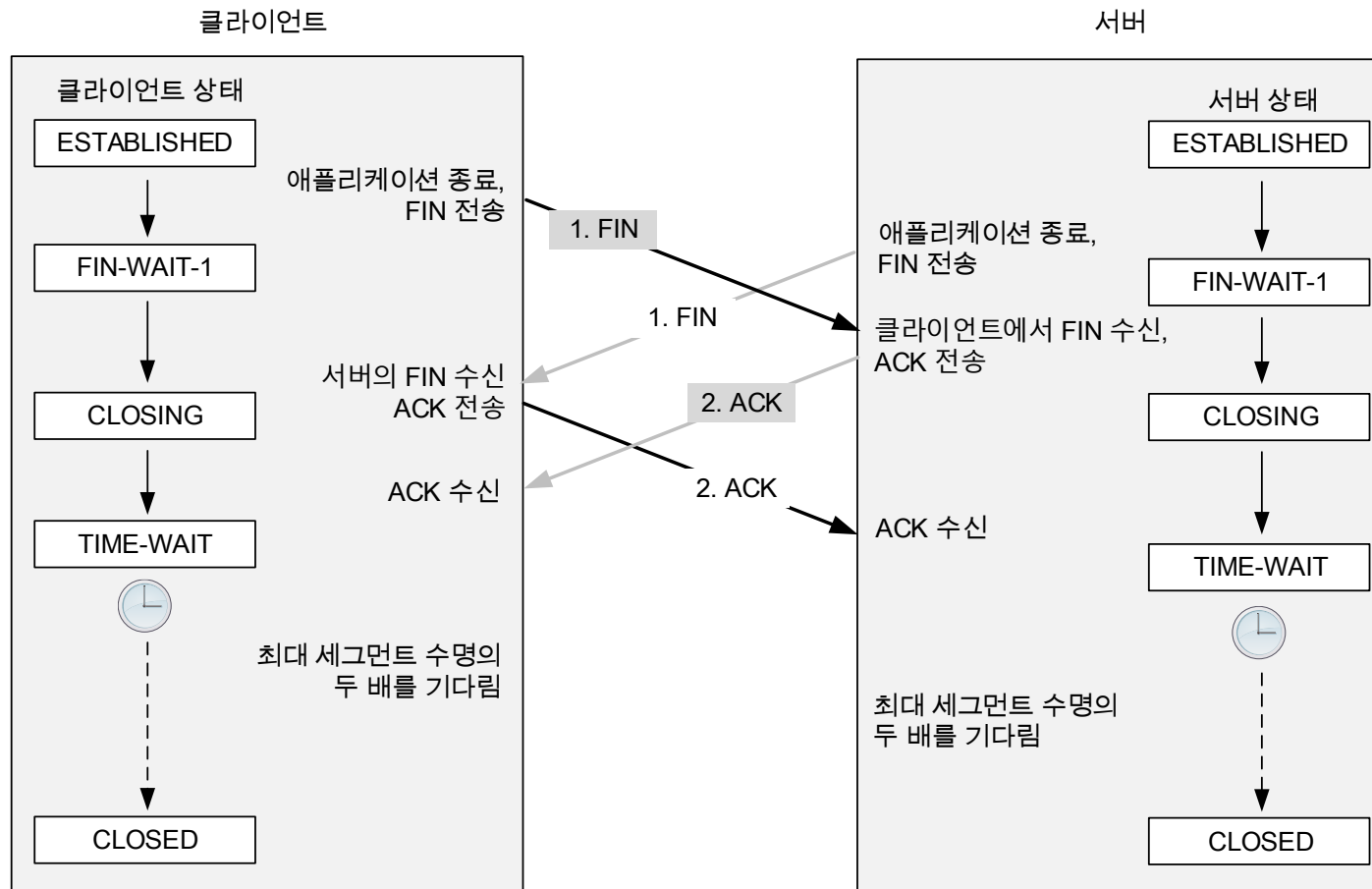
TCP 기본 동작

- TCP 연결 종료
 - 연결 종료 방법(1/2)
 - 일반적인 연결 종료 과정



TCP 기본 동작

- TCP 연결 종료
 - 연결 종료 방법(2/2)
 - 동시 종료 과정



목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

TCP 세그먼트 포맷과 데이터 송신

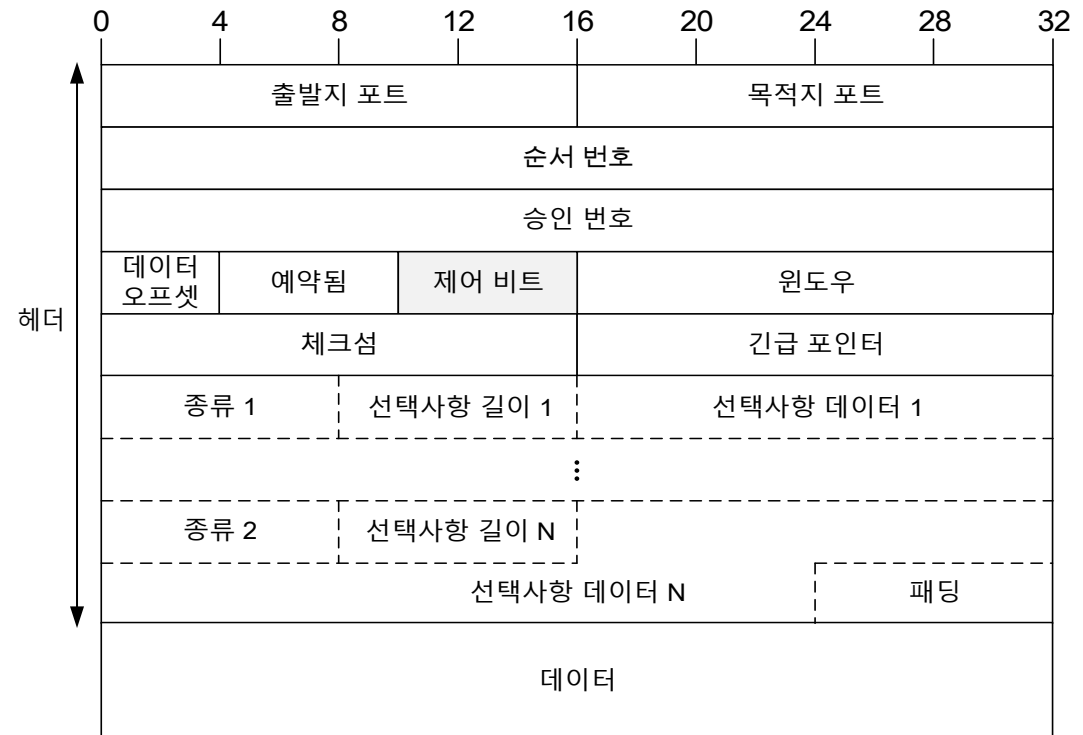
- TCP 세그먼트 포맷

- 목적

목적	설명
프로세스 주소 지정	<ul style="list-style-type: none">• 출발지와 목적지 장비에 있는 프로세스를 포트 번호로 식별
슬라이딩 윈도우 시스템 구현	<ul style="list-style-type: none">• 순서 번호, 승인 번호, 윈도우 크기 필드 이용
제어 비트와 필드 설정	<ul style="list-style-type: none">• 제어 기능 구현을 위한 비트와, 포인터나 데이터를 저장하는 필드• e.g, URG(Urgent), ACK(Acknowledge), PSH(Push), RST(Reset), SYN(Synchronize), FIN(Finish)
데이터 송신	<ul style="list-style-type: none">• 장비 간 실제로 송신되는 바이트 단위의 데이터
다양한 기능 구현	<ul style="list-style-type: none">• 데이터 보호를 위한 체크섬과 연결 수립을 위한 선택 사항 필드 이용

TCP 세그먼트 포맷과 데이터 송신

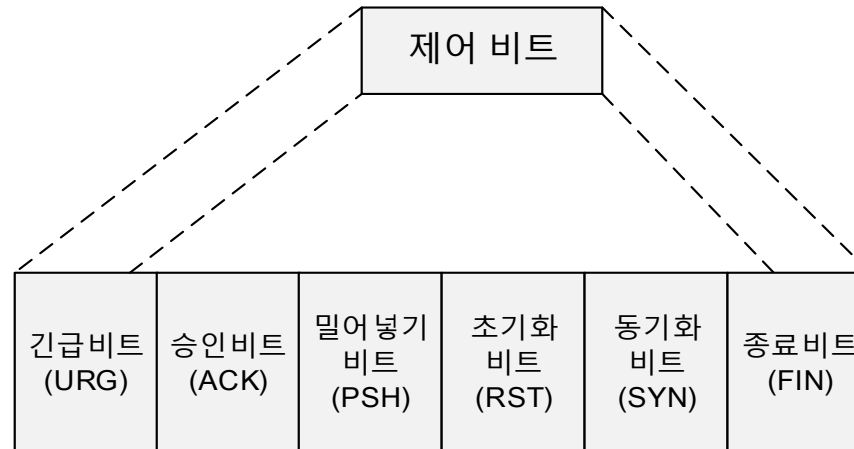
• TCP 세그먼트 포맷



필드명	크기(바이트)	설명
데이터 오프셋	½(4비트)	• 헤더 길이 표현, 세그먼트 시작 위치 표현
윈도우	2	• 한 번에 수신 가능한 바이트 수, 수신 윈도우 버퍼 크기
체크섬	2	• 프로토콜 헤더와 데이터에 대한 오류 검사
긴급 포인터	2	• URG 제어 비트가 1로 설정된 경우, 우선 순위 데이터 송신을 나타내는 URG 제어 비트의 마지막 바이트 순서 번호
선택사항	가변적	• 최대 세그먼트 크기 값, 윈도우 크기, 대체 체크섬
패딩	가변적	• 선택사항 필드가 32비트의 배수가 되도록 0으로 채움

TCP 세그먼트 포맷과 데이터 송신

• TCP 세그먼트 포맷



제어 비트 하위 필드명	크기(바이트)	설명
긴급(URG)	1	<ul style="list-style-type: none">우선 순위 높은 데이터가 있으므로 우선 처리해주어야 함을 알림
승인(ACK)	1	<ul style="list-style-type: none">세그먼트가 승인을 포함함을 알림승인 번호(상대 장비의 다음 순서 번호) 포함
밀어넣기(PSH)	1	<ul style="list-style-type: none">송신 장비가 TCP 밀어넣기 기능을 사용함송신 장비의 데이터를 수신한 즉시 처리해야 함
초기화(RST)	1	<ul style="list-style-type: none">송신 장비의 문제 발생으로 연결을 초기화해야 함
동기화(SYN)	1	<ul style="list-style-type: none">순서 번호(송신 장비의 ISN) 동기화 및 연결 수립을 요청함
연결 종료(FIN)	1	<ul style="list-style-type: none">송신 장비가 연결 종료를 요청함

TCP 세그먼트 포맷과 데이터 송신

- TCP 세그먼트 포맷

- TCP 가상 헤더(Pseudo Header)

- 체크섬 계산을 위해 생성되는 필드

- 송신된 자료의 무결성을 보호하는 방식

- 송신자는 데이터 바이트의 합을 데이터 스트림과 함께 송신

- 수신자는 데이터 바이트의 합을 검사하여 오류 발생 여부 확인

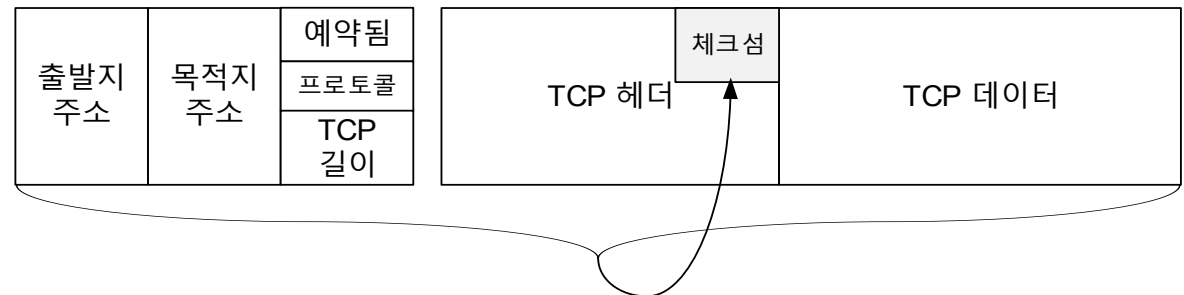
- 가상 헤더 포맷



가상 헤더

TCP 세그먼트

- 체크섬 계산 과정



TCP 세그먼트 포맷과 데이터 송신

- TCP 세그먼트 포맷
 - TCP 가상 헤더(Pseudo Header)
 - 가상 헤더 사용 장점
 - 잘못된 세그먼트 송신 문제 방지
 - 명시한 주소와 실제로 송수신한 주소가 다른 경우
 - 잘못된 프로토콜 문제 방지
 - 명시된 프로토콜이 아닌 프로토콜로 송신된 경우
 - 잘못된 세그먼트 길이 문제 방지
 - 세그먼트 일부가 유실된 경우, 출발지와 목적지가 사용하는 길이 값 불일치

TCP 세그먼트 포맷과 데이터 송신

- TCP 세그먼트 포맷
 - TCP 최대 세그먼트 크기(MSS, Maximum Segment Size)
 - 정의
 - 세그먼트의 데이터 필드에서 사용 가능한 최대 바이트 수
 - TCP 데이터(페이로드) 길이만을 의미
 - MSS 선택 시 고려 사항
 - 과부하 관리
 - TCP 헤더와 IP 헤더는 각각 20바이트 이상씩 사용
 - MSS가 너무 작은 경우, 대역폭을 비효율적으로 사용하게 됨
 - IP 단편화
 - 세그먼트는 최대 송신 단위(MTU, Maximum Transmission Unit) 이상 송신 불가능
 - IP 헤더와 TCP 헤더, TCP 데이터를 모두 포함하는 길이
 - MTU 이상의 세그먼트를 송신하려면 단편화를 해야 하므로 비효율적

TCP 세그먼트 포맷과 데이터 송신

- TCP 세그먼트 포맷
 - TCP 최대 세그먼트 크기(MSS)
 - 기본 MSS
 - 보통 세그먼트에서 단편화 없이 송신될 수 있는 최대 크기로 결정
 - TCP에서 사용하는 표준 MSS는 536바이트
 - TCP와 IP 헤더로 최소 40바이트를 사용
 - IP 네트워크의 최소 MTU는 576(536+40)바이트에서 시작
 - MSS 값 명시
 - SYN 메시지의 최대 세그먼트 크기 필드 이용하여 명시
 - 표준 MSS보다 크거나 작은 MSS를 사용하는 경우에 MSS 값 명시
 - e.g., IPsec을 사용하는 경우, AH 헤더나 ESP 헤더 추가로 인해 헤더 크기가 커지기 때문에 더 작은 MSS를 사용할 것임

TCP 세그먼트 포맷과 데이터 송신

- TCP 데이터 송신 방식: 슬라이딩 윈도우

- 전송 카테고리

카테고리	설명
전송 카테고리 1	전송했고 승인 받음
전송 카테고리 2	전송했지만 아직 승인 받지 못함
전송 카테고리 3	수신자는 준비됐지만 아직 전송하지 못함
전송 카테고리 4	수신자가 준비되지 않았고 전송하지도 못함

- 수신 카테고리

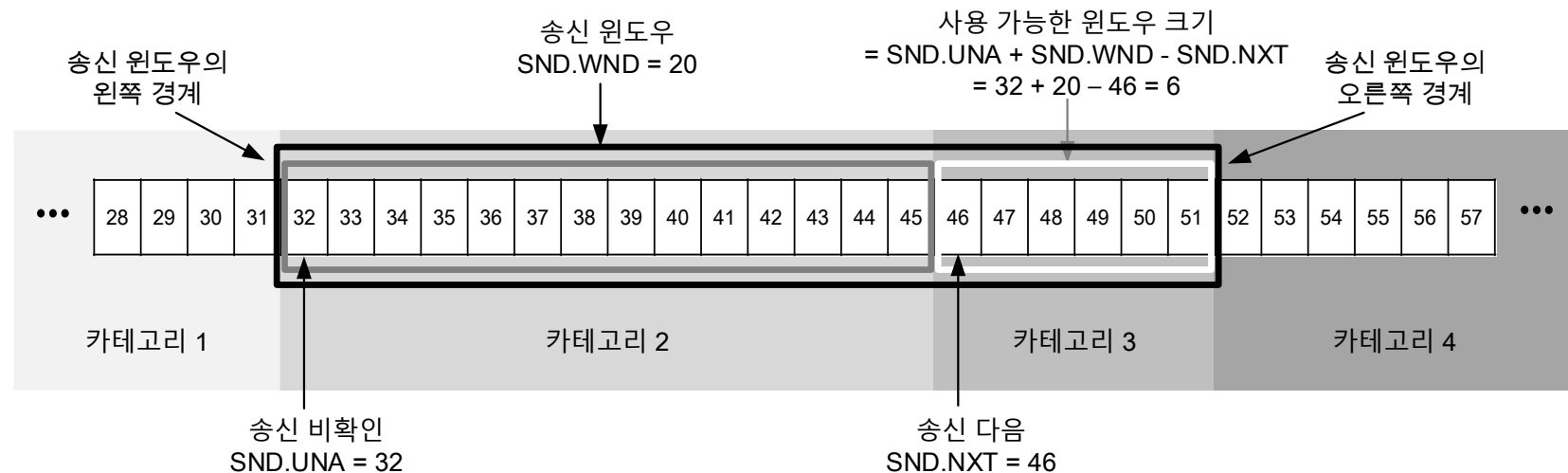
카테고리	설명
수신 카테고리 1+2	수신했고 승인함
수신 카테고리 3	수신자는 준비됐지만 아직 수신하지 못함
수신 카테고리 4	수신자가 준비되지 않았고 수신하지도 못함

TCP 세그먼트 포맷과 데이터 송신

• TCP 데이터 송신 방식: 슬라이딩 윈도우

• 송신(SND) 포인터

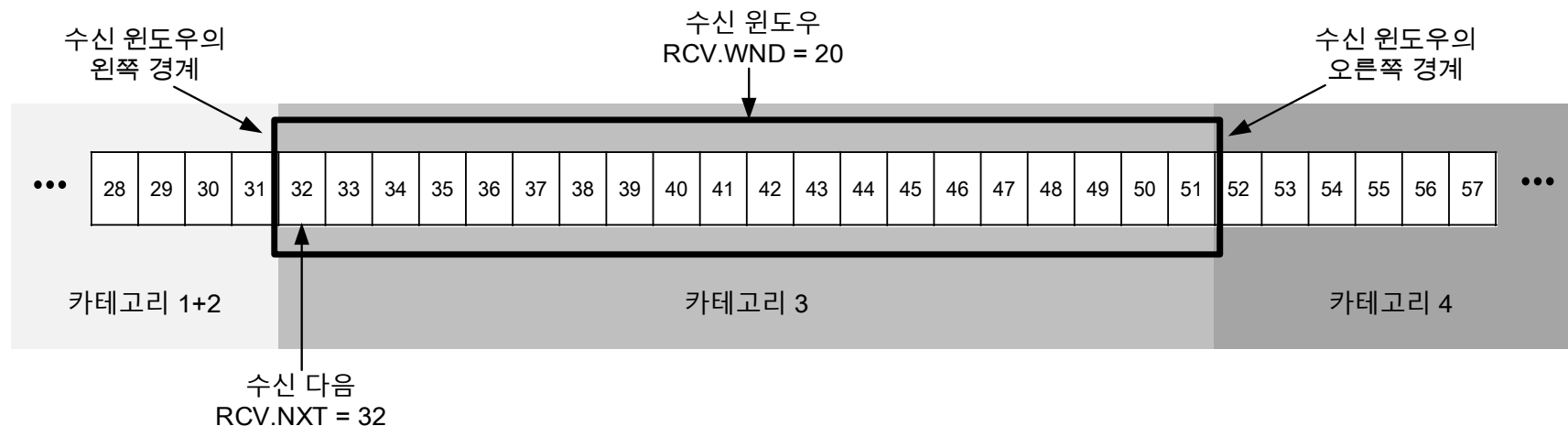
송신(SND) 포인터	설명
송신 비확인 (SND.UNA)	<ul style="list-style-type: none">• 송신했지만 아직 승인되지 않은 첫 번째 데이터 순서 번호• 전송 카테고리 2의 첫 바이트 가리킴• 이전 순서 번호는 모두 전송 카테고리 1
송신 다음 (SND.NXT)	<ul style="list-style-type: none">• 상대 장비에게 송신할 다음 바이트의 순서 번호• 전송 카테고리 3의 첫 바이트 가리킴
송신 윈도우 (SND.WND)	<ul style="list-style-type: none">• 송신 윈도우의 크기이자 승인 없이 보낼 수 있는 총 바이트 수• 송신 비확인(SND.UNA)과 더하면, 전송 카테고리 4의 첫 바이트 표시 가능



TCP 세그먼트 포맷과 데이터 송신

- TCP 데이터 송신 방식: 슬라이딩 윈도우
- 수신(RCV) 포인터

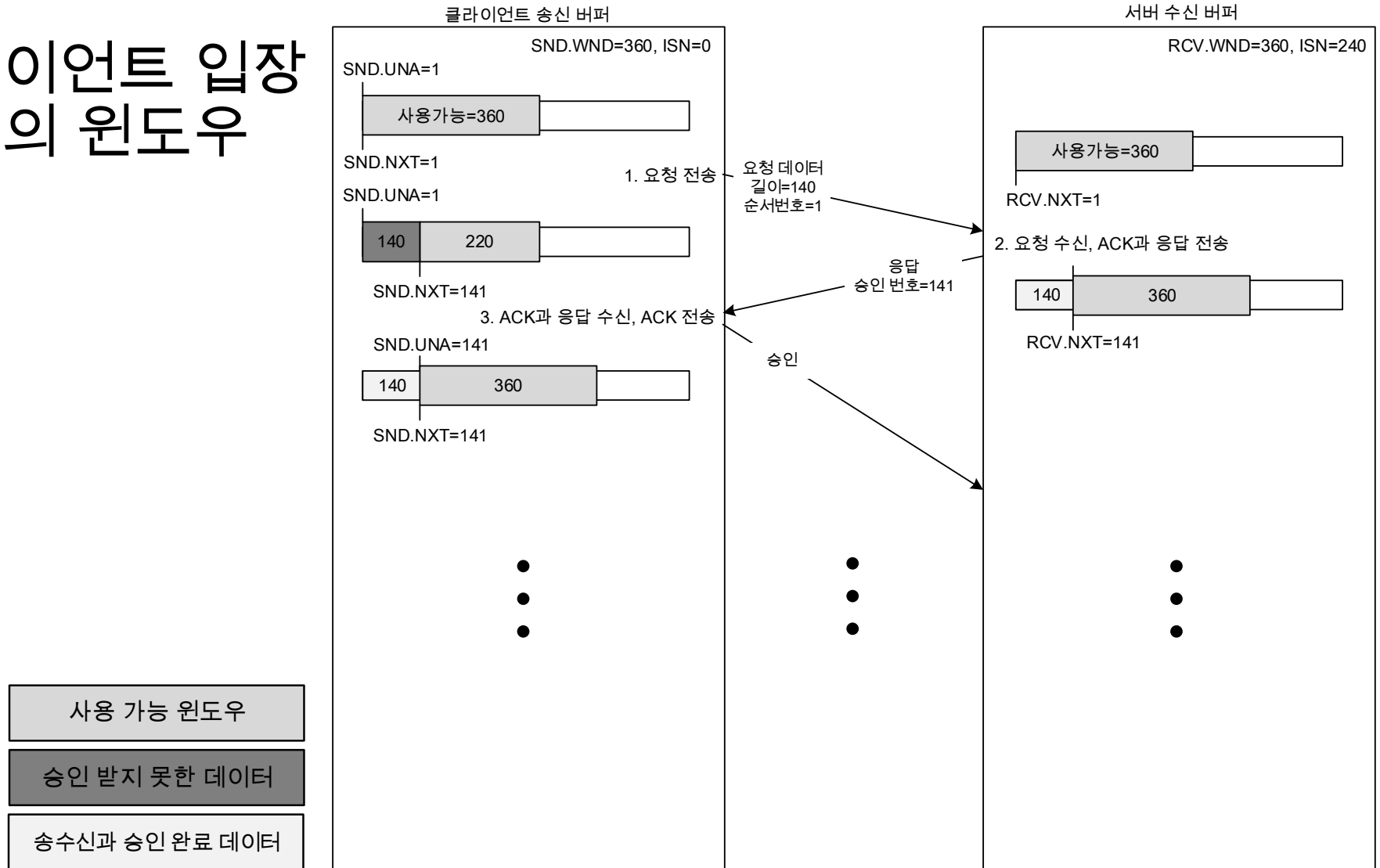
수신(RCV) 포인터	설명
수신 다음 (RCV.NXT)	<ul style="list-style-type: none">• 상대 장비에서 수신하려는 데이터의 다음 바이트 순서 번호• 수신 카테고리 3의 첫 바이트를 가리킴
수신 윈도우 (RCV.WND)	<ul style="list-style-type: none">• 수신 윈도우의 크기이자 장비가 한 번에 수신하길 원하는 바이트 수• 수신 다음(RCV.NXT)과 더하면, 수신 카테고리 4의 첫 바이트 표시 가능



TCP 세그먼트 포맷과 데이터 송신

• TCP 데이터 송신 방식: 슬라이딩 윈도우

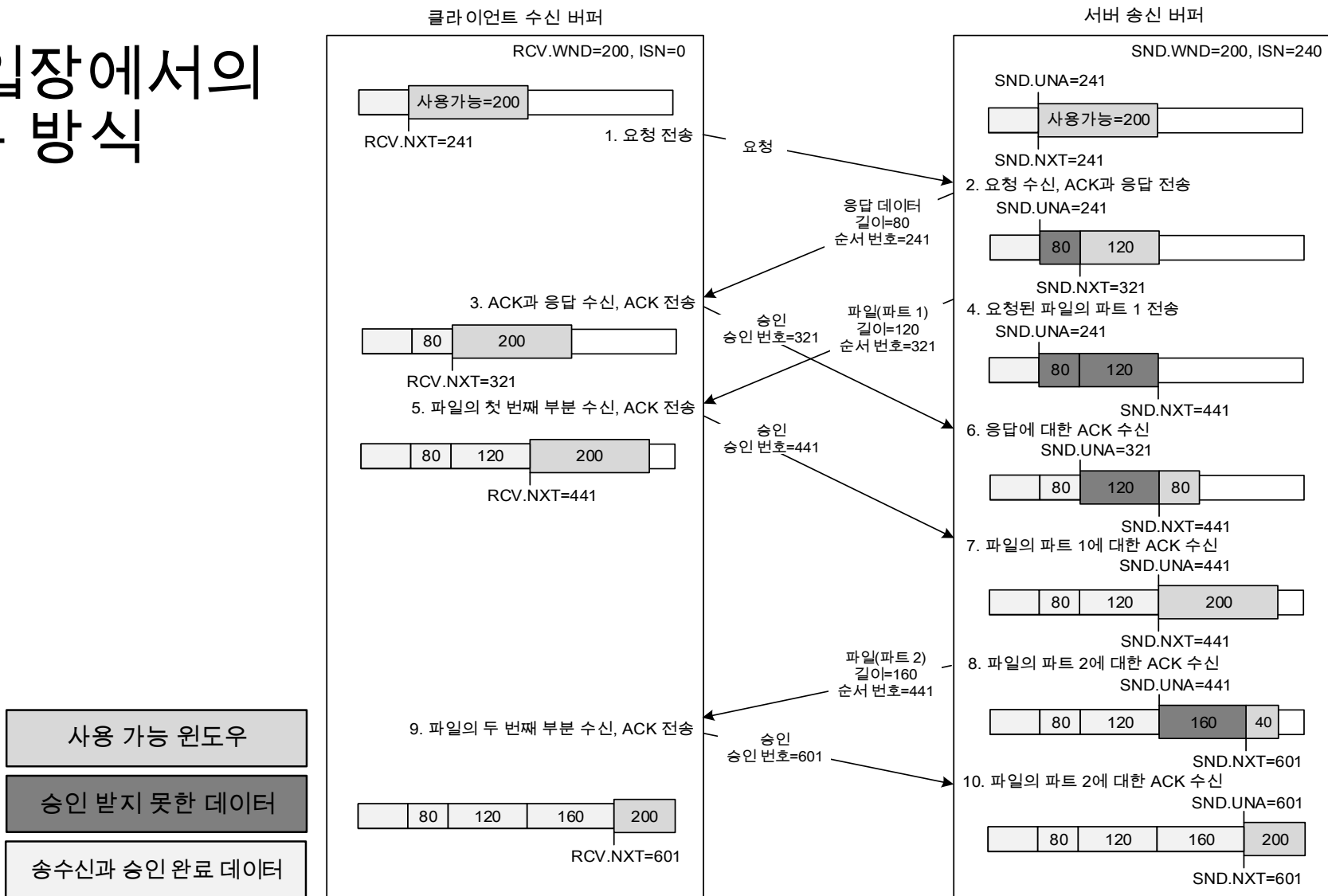
• 클라이언트 입장에서 의 윈도우



TCP 세그먼트 포맷과 데이터 송신

• TCP 데이터 송신 방식: 슬라이딩 윈도우

• 서버 입장에서의 윈도우 방식



목 차

- 보충
 - ICMPv4 오류 메시지 유형
 - ICMPv4 정보 제공 메시지 유형
- TCP/IP 전송 계층 프로토콜
 - TCP 개요
 - TCP 원리와 일반 동작
 - TCP 기본 동작: 연결 수립, 관리와 종료
 - TCP 세그먼트 포맷과 데이터 송신
 - TCP 신뢰성과 흐름 제어 기능

TCP 신뢰성과 흐름 제어 기능

- TCP 세그먼트 재전송

- 재전송 과정

1. 재전송 큐(Queue)에 배치, 타이머 시작

- 세그먼트 전송 시, 복사본을 재전송 큐라는 데이터 구조에 삽입
 - FIFO(First In First Out) 구조로 데이터를 저장하는 형식의 스택 구조
- 큐에 삽입 시 재전송 타이머 시작

2. 승인 처리

- 타이머 만료 전에 승인이 오는 경우, 큐에서 세그먼트 제거
 - 세그먼트가 지연되거나 버려지는 경우에 재전송 타이머가 만료되었다고 함

3. 재전송 시간 만료

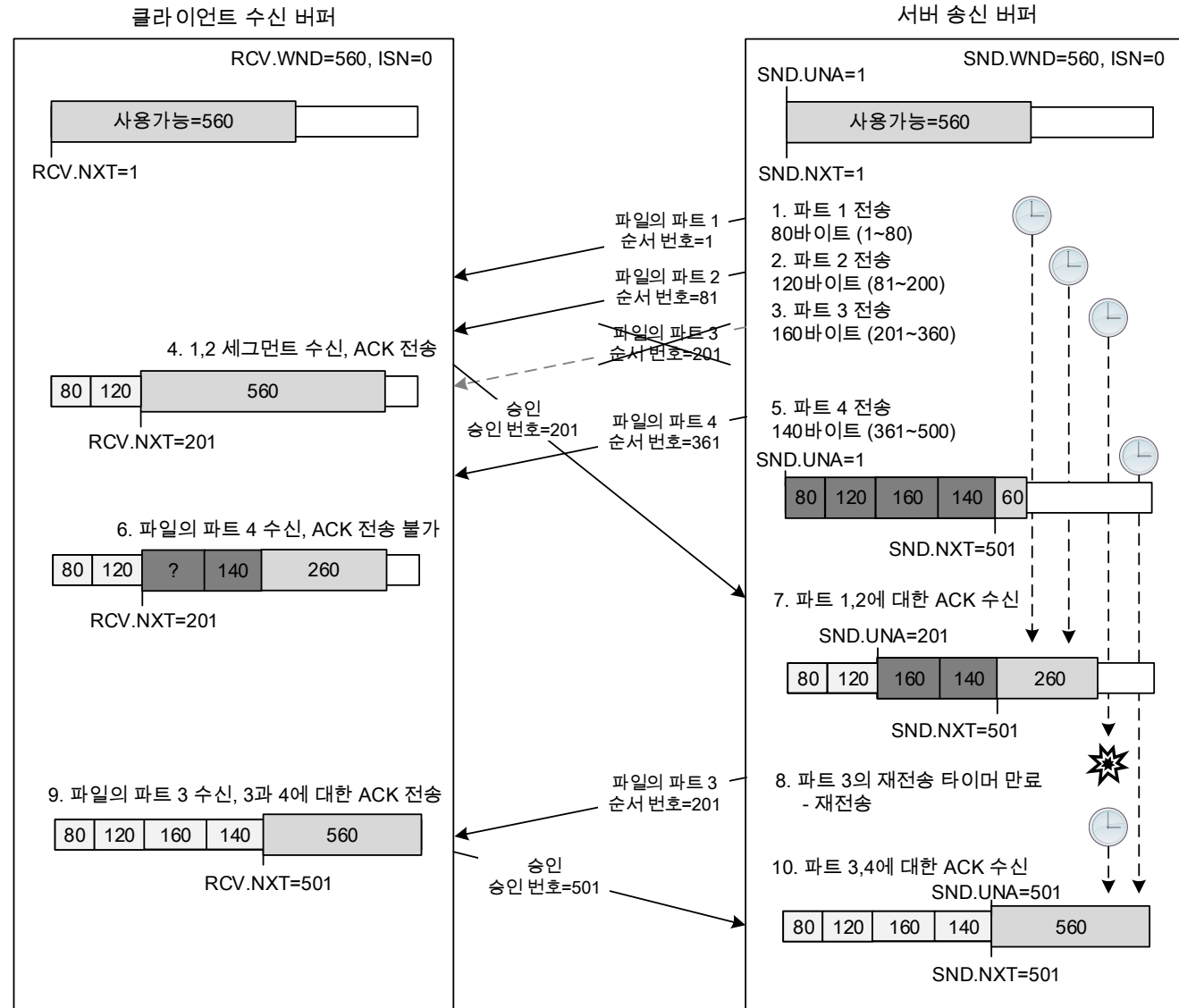
- 타이머 만료 전에 승인이 오지 않는 경우, 재전송 시간 만료로 인해 세그먼트 자동 재전송

TCP 신뢰성과 흐름 제어 기능

• TCP 세그먼트 재전송

• 재전송 과정

- e.g., 서버가 연속해서 네 개의 세그먼트를 송신한 경우



TCP 신뢰성과 흐름 제어 기능

- TCP 재전송 처리 방식

- 비연속적 승인 처리

- 시간 만료된 세그먼트만 재전송

- 여러 개의 세그먼트가 사라진 경우, 대기 시간 증가

- e.g., 세그먼트 20개가 모두 사라진 경우

- 모든 세그먼트가 각각 시간이 만료되어 재전송될 때까지 대기해야 함

- 세그먼트 1이 시간 만료되어 재전송되고, 이에 대한 ACK를 받아도 세그먼트 2가 시간 만료되어 재전송될 때까지 기다려야 함

- 승인 받지 못한 모든 세그먼트 재전송

- 시간 만료된 세그먼트와 승인 받지 못한 세그먼트 재전송

- 필요하지 않은 경우에도 재전송하는 문제 존재

- e.g., 세그먼트 20개가 모두 사라진 경우

- 첫 번째 세그먼트만 사라지고 19개는 잘 도착했더라도 첫 번째 세그먼트 때문에 모든 세그먼트의 승인을 받지 못했기 때문에 전체적으로 재전송

TCP 신뢰성과 흐름 제어 기능

- TCP 재전송 처리 방식

- 선택적 승인(SACK, Selective Acknowledgment)

- 비연속적 승인 처리의 문제를 개선한 방식
- 연결 수립 시 선택적 승인 허용 선택 사항 사용
 - 수신했으나, 아직 승인하지 않은 세그먼트 범위에 대한 목록

- 동작 과정

1. 각 장비는 재전송 큐를 수정하여 세그먼트가 선택적으로 승인된 경우, SACK 비트를 1로 설정
 - e.g., 서버가 파트 1,2,3,4를 송신했으나, 파트 3이 전송 중 유실된 경우, 클라이언트는 파트 1,2에 대한 승인을 보내면서 파트 4에 대한 SACK 비트를 1로 설정하여 전송함으로써, 서버에게 파트 4는 재전송할 필요없음을 알림
2. 특정 세그먼트 재전송 시, 세그먼트 중 SACK 비트가 1이 아닌 모든 세그먼트 재전송

TCP 신뢰성과 흐름 제어 기능

- TCP 재전송 처리 방식
 - RTT(Round-Trip Time) 계산에 기반한 적응형 재전송
 - 왕복 시간(RTT)
 - 송신된 세그먼트가 도착하여 승인을 되돌려 보내는데 걸리는 시간
 - 재전송 타이머 사용을 위해 예상 RTT 계산 필요
 - 각 데이터그램마다 RTT는 다르기 때문에 RTT 평균치 필요
 - RTT 평균 추정 계산
 - 새 $RTT = a * \text{기존 RTT} + ((1 - a) * \text{측정한 RTT})$
 - a 는 0에서 1까지의 값을 가지는 가중치, 보통 a 는 0.125
 - a 가 0에 가까운 경우, RTT 변화가 증가하여 새 RTT 값이 자주 바뀔 수 있음
 - 한계점
 - 모호한 승인(Acknowledgment Ambiguity)
 - 원래 세그먼트에 대한 승인인지, 재전송 세그먼트에 대한 승인인지 알 수 없는 현상

TCP 신뢰성과 흐름 제어 기능

- TCP 재전송 처리 방식

- 한계점 개선 방안

- 칸(Karn)의 알고리즘

- 재전송된 세그먼트에 적용되는 타이머 시간 계산과 평균 RTT 계산을 분리하는 알고리즘

- 세그먼트 전송 실패 시, RTT 값을 두 배로 하여 사용하는 알고리즘

- 타이머 백오프(Backoff) 방식 사용

- 통신할 때 일정 시간 대기 후, 다시 시도하는 방식
 - 세그먼트 재전송 시 타이머를 백오프로 두 배 증가시켜 재전송 시간 설정
 - 타이머 값은 재전송 성공할 때까지 증가하지만 일정 한도 존재
 - 세그먼트가 빠르게 재전송되는 상황을 방지할 수 있음

TCP 신뢰성과 흐름 제어 기능

- TCP 흐름 제어

- TCP 윈도우 크기 조절

- 상대 장비에서 들어오는 데이터 흐름을 조절하는 방법
 - 윈도우 크기
 - 송신 전, 상대 장비로부터 받은 데이터를 저장할 수 있는 바이트 수
 - e.g., 서버 윈도우 크기가 360인 경우, 서버는 360바이트 이상 받지 않음
 - 윈도우 크기 감소
 - 송신 장비가 전달할 수 있는 데이터 양을 감소시킴
 - 송신 윈도우에서 사용 가능 윈도우 크기가 0이 될 때까지 데이터를 송신해줌

TCP 신뢰성과 흐름 제어 기능

- TCP 바보 윈도우 증후군(SWS, Silly Window Syndrome)
 - 정의
 - 세그먼트 크기가 헤더 크기보다 작은 경우, 전송 시 최소 40바이트의 헤더를 붙여주어야 하는 비효율적인 현상
 - 회피 알고리즘
 - 수신자 SWS 회피
 - 수신 윈도우 크기가 최소 단위보다 작은 경우, 송신 측에 보내지 않고 대기
 - 최소 단위는 MSS나 버퍼의 절반 중 작은 것으로 결정
 - 수신 장비는 윈도우 크기를 고정해두었다가, 공간이 생기는 경우에는 윈도우 크기를 최소 단위만큼 증가시켜야 함
 - 송신자 SWS 회피와 네이글(Nagle)의 알고리즘
 - 데이터 크기가 일정 크기가 되는 경우에만 데이터를 전송하여 네트워크 효율성을 올리는데 사용

TCP 신뢰성과 흐름 제어 기능

- TCP 혼잡과 제어 알고리즘
 - 혼잡(Congestion)
 - 세그먼트 송신 속도가 느려지거나 버려지는 경우를 의미
 - 계속적인 재전송에 의해 혼잡 붕괴(Congestion Collapse) 현상 발생 가능
 - e.g., UDP와 TCP를 혼용할 경우 생기는 충돌 현상
 - 데이터 유실이 생겨도 빠르게 많이 보내는 UDP와 순서대로 데이터를 보내며 데이터 유실 시 속도 조절하는 TCP의 혼용
 - 세그먼트 송신 후 승인 받지 못한 비율로 장비 간 네트워크의 혼잡 정도를 알 수 있음

TCP 신뢰성과 흐름 제어 기능

- TCP 혼잡과 제어 알고리즘
 - 혼잡 제어 알고리즘 (1/2)
 - 느린 시작(Slow Start) 알고리즘
 - 혼잡 윈도우 크기를 임계점에 도달할 때까지 지수적으로 증가시키는 알고리즘
 - 임계점에 도달하면 혼잡 회피 단계로 이동
 - 혼잡 현상이 발생하면 혼잡 윈도우 크기를 1로 감소시킴
 - 느리게 시작하지만 증가 속도는 점점 빨라짐
 - 혼잡 회피(Congestion Avoidance) 알고리즘
 - 혼잡 상태가 감지될 때까지 윈도우 크기를 1씩 증가시키는 알고리즘
 - 승인을 세 번 이상 받는 경우
 - 큰 값으로부터 시작하지만 증가 속도는 1로 일정함

TCP 신뢰성과 흐름 제어 기능

- TCP 혼잡과 제어 알고리즘

- 혼잡 제어 알고리즘 (2/2)

- 빠른 재전송(Fast Retransmit) 알고리즘

- 재전송 큐 과정 생략 후 사라진 세그먼트를 재전송하는 알고리즘
 - 타이머 만료와 관계 없이 해당 세그먼트만 재전송이 가능하여 빠른 전송률 유지
 - 승인을 세 번 이상 받는 경우, 세그먼트를 재전송한 후 혼잡 윈도우 크기를 줄이는 알고리즘

- 빠른 회복(Fast Recovery) 알고리즘

- 승인을 세 번 이상 받는 경우, 혼잡 윈도우 크기를 절반으로 줄인 후 송신 속도를 1씩 증가시키는 알고리즘
 - 혼잡 회피 알고리즘 이용

Thanks!

김 지 혜 (jihye@pel.sejong.ac.kr)