

네트워크란?

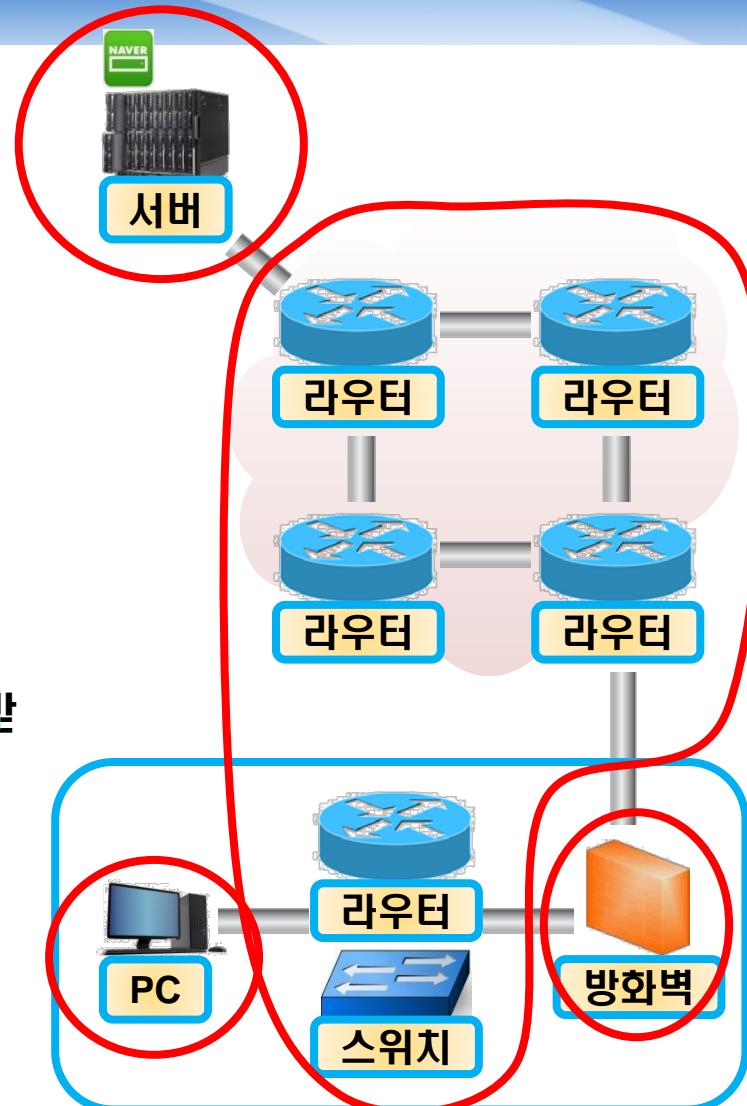
# 네트워크 개요(1/4)

## □ 네트워크(Network)의 정의

- 정보의 공유를 위해서 통신망을 이용하여 단말장비들을 연결해놓은 시스템(net + work)

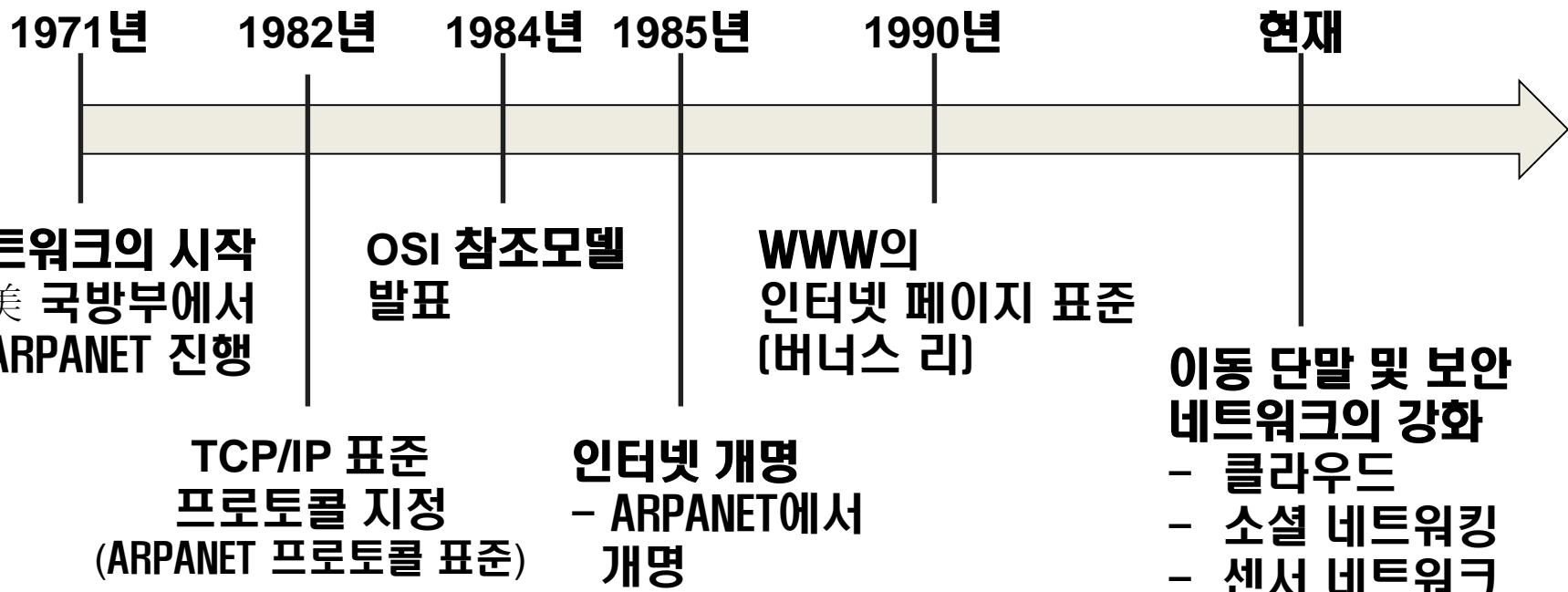
## □ 네트워크 구성요소

- 단말장비 : 데이터를 생산, 저장, 사용하는 개체  
ex) 서버, PC
- 교환장비 : 단말장비들이 데이터를 서로 주고 받을 수 있도록 연결해주는 중간장비  
ex) 라우터, 스위치
- 네트워크 관리장비 : 기업, 학교 등에서 내부 네트워크를 관리하고 보호하기 위한 시스템  
ex) 방화벽



# 네트워크 개요(2/4)

## □ 네트워크의 발전



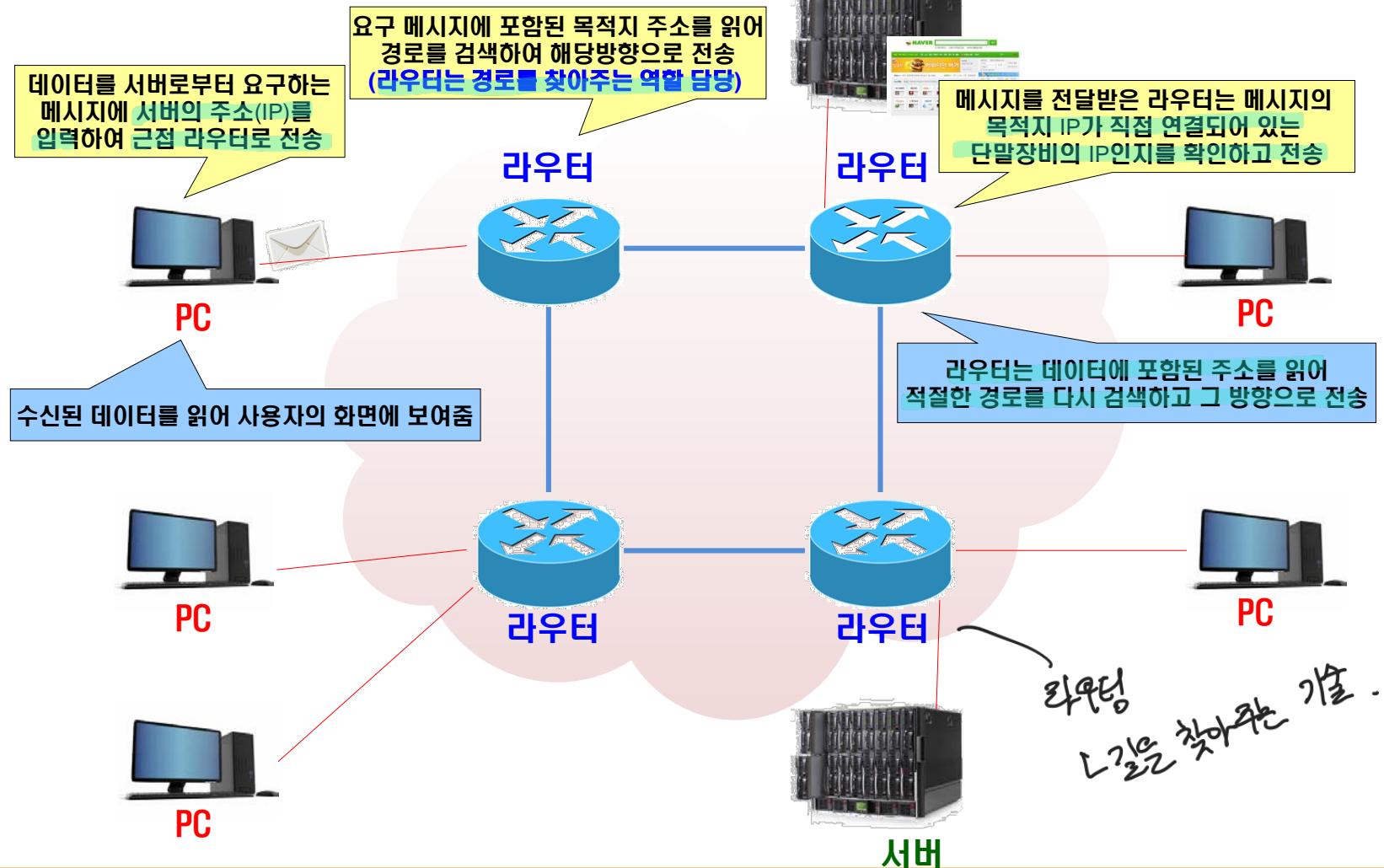
### ※ 약어 참고

- ARPANET (Advance Research Projects Agency Network)
- WWW (World Wide Web)
- TCP/IP (Transmission Control Protocol / Internet Protocol)
- OSI model (Open Systems Interconnection model)

# 네트워크 개요

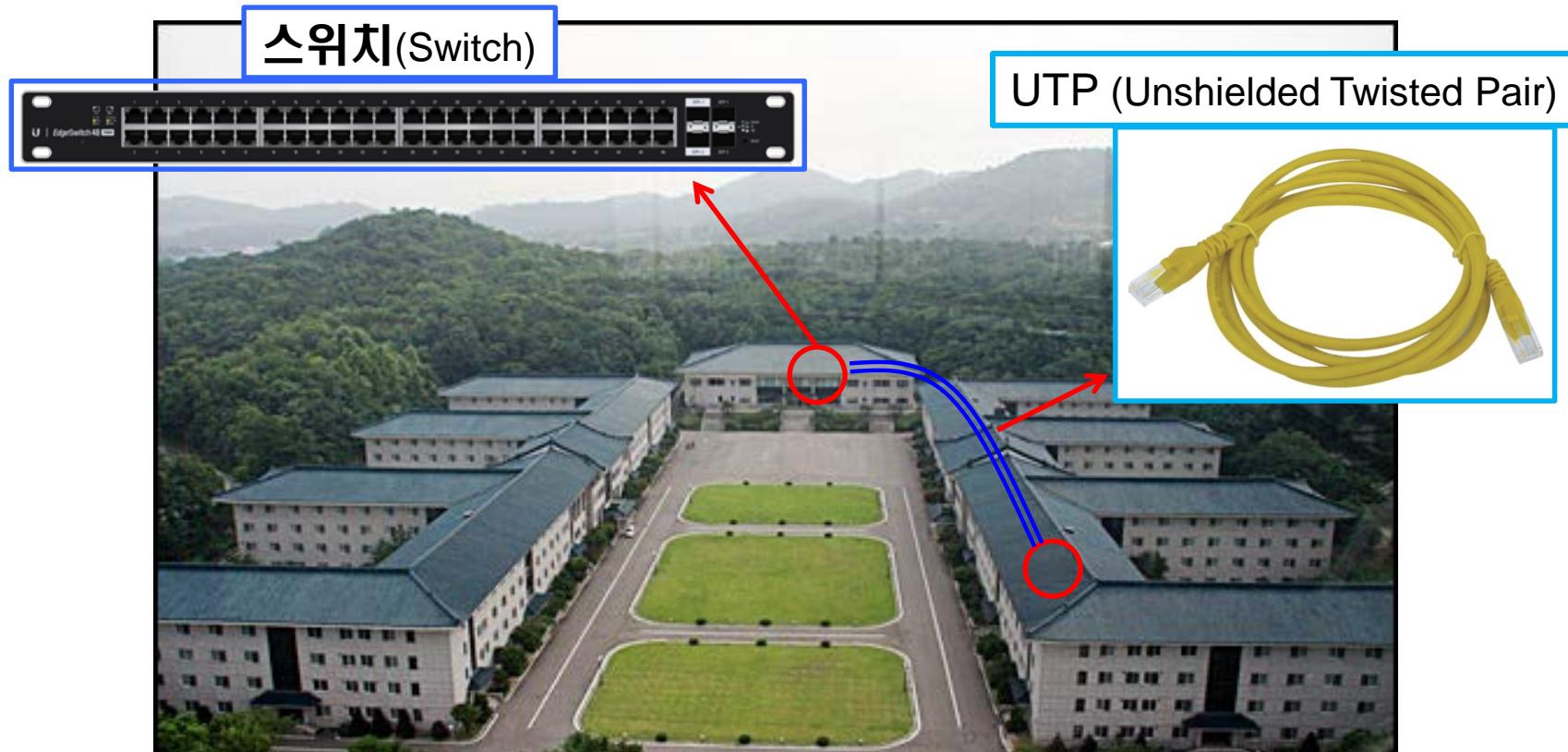
서버는 요구메시지를 확인하여 요구되는 데이터를 생산하고, 다시 목적지 주소(요구한 PC의 주소)를 기입하여 다시 인접 라우터에 전송

## □ 네트워크의 동작 (예)



# 네트워크 개요(4/4)

## □ 네트워크 구성





# 네트워크 프로토콜(1/3)

## □ 네트워크 프로토콜의 정의

- 데이터에 포함되는 주소의 표기, 데이터의 크기, 데이터의 전송속도 등  
상호 약속된 규칙

## □ 네트워크 프로토콜의 개발/발전

- 네트워크는 단일 장비가 아닌 장비들이 결합되어 있는 시스템으로서, 시스템이 제대로 동작하기 위해 각 장비들이 따라야 하는 규칙 필요
  - ex) 편지에 주소를 적을 때 보내는 사람과 받는 사람의 주소를 적는 위치가 정해져 있음

☞ 이 프로토콜들을 효과적으로 표준화하기 위하여 네트워크 동작에 따른  
**계층화된 모델이 필요**

\* ISO(International Standard Organization)에서 OSI(open System Interconnection) 7계층 제시('84)

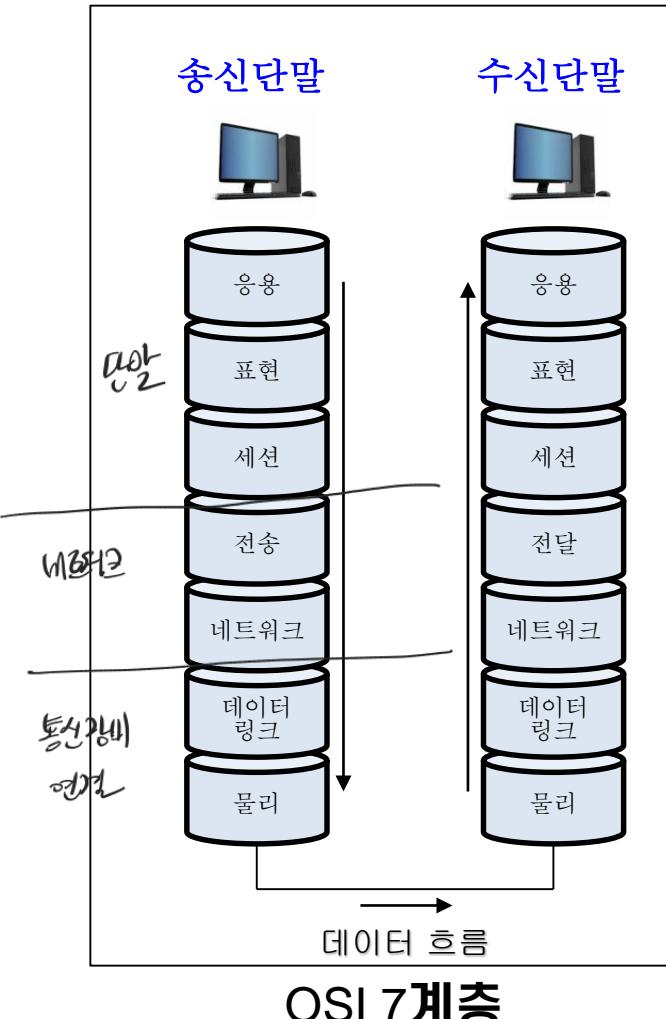
# 네트워크 프로토콜(2/3)

## □ OSI 7계층 모델

- 최상위 계층인 응용계층에서 데이터 전송이 요구되면, 순차적으로 한계단씩 전달되어 물리적인 전송 매체를 통하여 전송되어짐
- 수신단에서는 반대 과정으로 가장 하위 계층에서 상위 계층으로 전송되어 데이터 수신이 이뤄짐

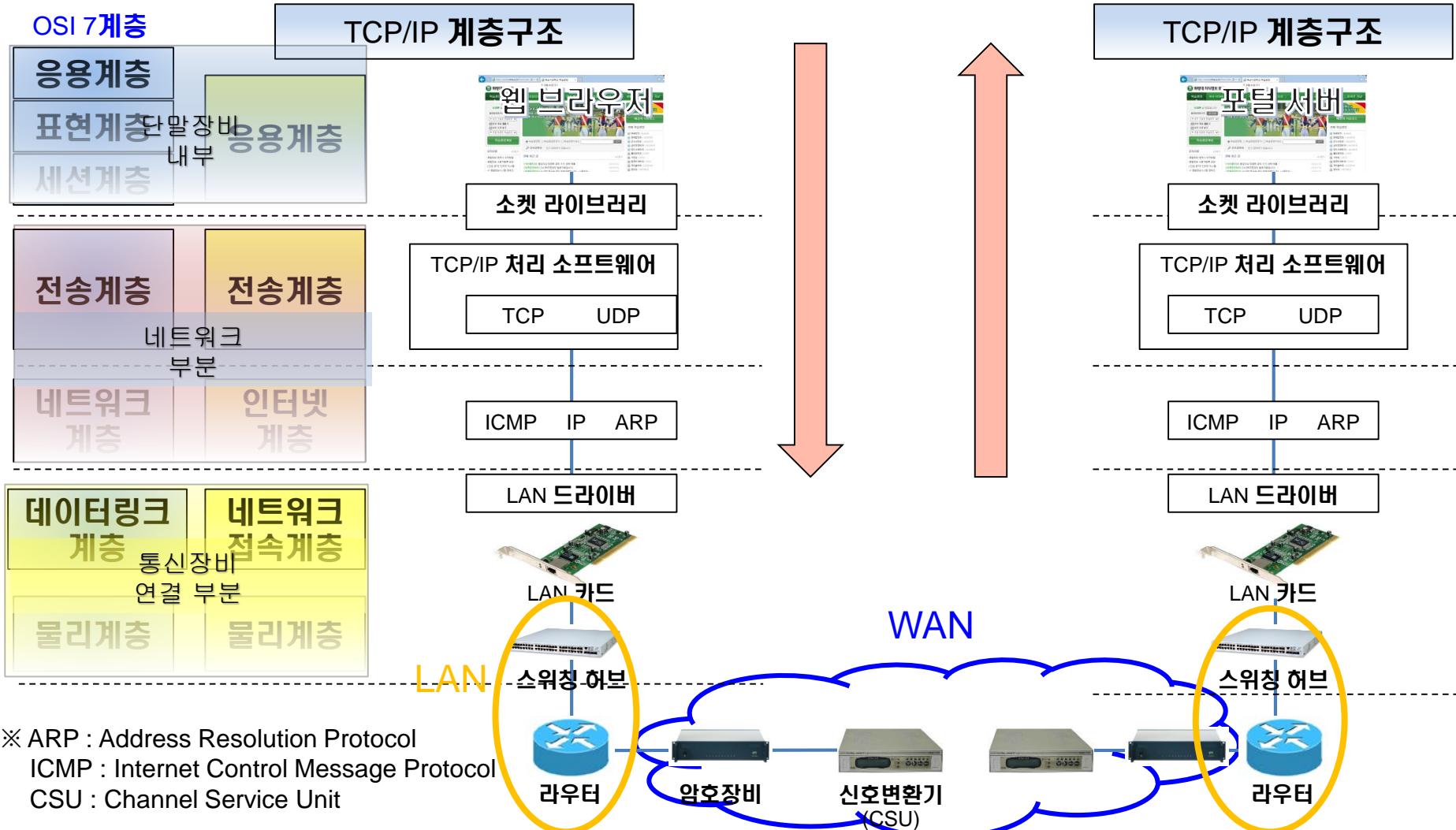
## □ 계층화(Layer)의 장점

- 각 계층이 독립화되어 문제해결 및 관리 용이
- 새로운 기술 접목시 수정이 용이



# 네트워크 프로토콜(3/3)

## □ TCP/IP 모델





# 네트워크 계층(1/2)

## □ 네트워크 계층

- 네트워크 계층(Network Layer)는 송신자로부터 수신자까지 데이터를 전송하기 위한 적절한 경로를 찾는 과정, 즉 라우팅 문제를 처리함
- 라우팅을 수행하기 위하여 각 단말 및 노드들은 주소를 가져야 하는데, 이 주소를 IP(Internet Protocol) 주소라고 함

### ※ 주소의 종류

IP(Internet Protocol) 주소 : 총 32개의 비트로 구성된 주소체계로서, 0~255사이의 4개의 십진수를 사용 (0.0.0.0 ~ 255.255.255.255)

MAC(Media Access Control) 주소 : 네트워크 어댑터 주소로서 네트워크 어댑터마다 부여된 준고유 식별자 (생산시 부여)

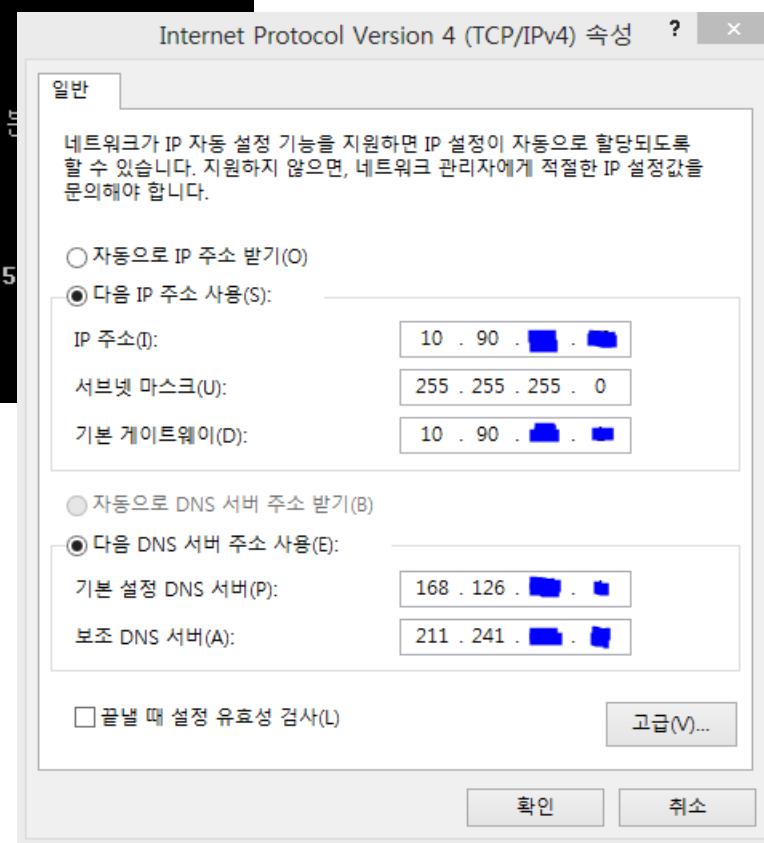
※ IP 주소는 소프트웨어적인 주소이며, MAC 주소는 하드웨어에 부여되는 고유번호임  
☞ IP와 MAC 주소는 상호 보완적 역할

# 네트워크 계층(2/2)

## □ IP 주소 및 MAC 주소

이더넷 어댑터 이더넷 2:

연결별 DNS 접미사..... : Marvell Yukon 88E8056 PCI-E Gigabit Ethernet Co  
ntroller  
물리적 주소..... : 00-21-85-53-50-66  
DHCP 사용..... : 아니요  
자동 구성 사용..... : 예  
링크-로컬 IPv6 주소..... : fe80::f5dd:2102:81e9:135c%19<기본>  
IPv4 주소..... : 10.90.1.10 <기본 설정>  
서브넷 마스크..... : 255.255.255.0  
기본 게이트웨이..... : 10.90.1.1  
DHCPv6 IAID..... : 401894660  
DHCPv6 클라이언트 DUID..... : 00-01-00-01-19-DD-57-5D-00-21-85  
DNS 서버..... : 168.126.1.1  
211.241.1.1  
Tcpip를 통한 NetBIOS..... : 사용



MAC: oct (8)

DHCP: 유동 IP 받기 (관리)

DNS: Domain Name Server → Domain ↔ IP 매핑

게이트웨이: 리우터 포트에 할당되어 있는, 이 네트워크를 지나기  
위한 경로이자 고정적인 값 설정. → 리우터 포트에  
할당되어 있는 주소.

# IP 주소체계(1/8)

## □ IP주소

- 왜 필요한가? 인터넷에서 정보를 정확하게 전달하기 위한 유일한 식별자
- ✓ IP 관리기구 : ICANN(전세계) / APNIC(아시아) / KISA(한국)

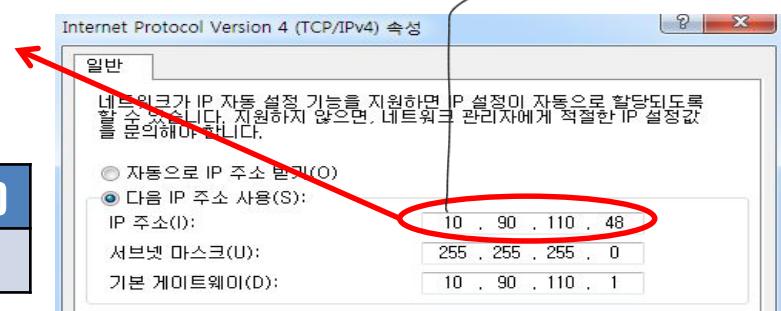


- ✓ IPv4 구성 : 32개 비트를 8개 비트(1바이트)씩 4개의 Octet으로 구성

예 ) IP 주소 : 10 . 90 . 110 . 48

\* DDN(Dotted-Decimal Notation)로 표현

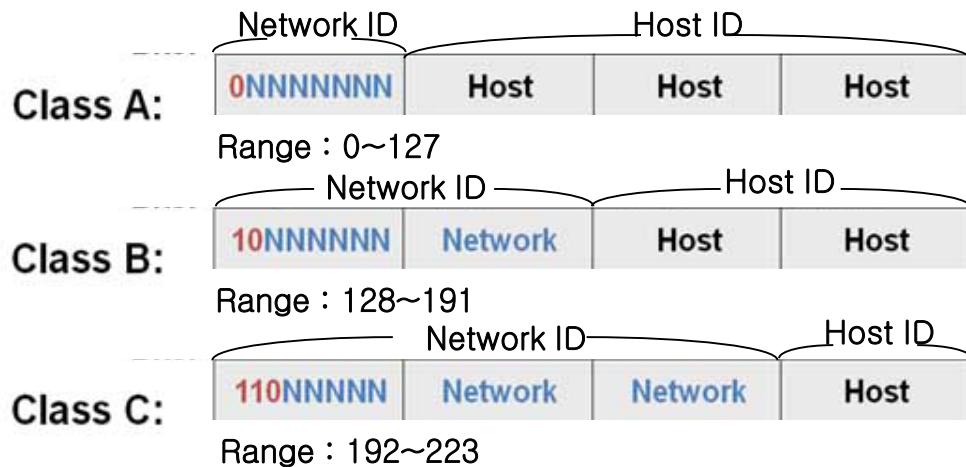
이진수	00001010	01100100	01101110	00110000
십진수	10	90	110	48



# IP 주소체계[2/8]

## □ IP 주소체계

- IP 주소는 네트워크 ID와 호스트 ID로 구성되어 있음
- 네트워크 ID와 호스트 ID의 구분은 클래스[A~C]에 따라 달라짐



- \* Class D : 멀티캐스트용 (Range : 224 ~ 239)
- \* Class E : 실험용 및 예비 (Range : 240 ~ 255)

A클래스는 첫번째 비트가 0으로 시작  
호스트 수 :  $2^{24} - 2 = 16,777,214$  개

B클래스는 첫번째 비트가 10으로 시작  
호스트 수 :  $2^{16} - 2 = 65,534$  개

C클래스는 첫번째 비트가 110으로 시작  
호스트 수 :  $2^8 - 2 = 254$  개

\* 2개를 빼주는 이유:  
Host ID가 모두 0인 경우와 Host ID가 모두 1인 경우는  
네트워크 ID와 브로드 캐스팅을 목적으로 사용되기 때문

※ 0: 그 네트워크 자체를 가리기  
1: 네트워크 ID → 브로드캐스팅  
∴ 2를 빼줌.



# IP 주소체계[3/8]

## □ IP 주소체계

- 클래스에 따른 IP 주소 분류

클래스	시작비트	네트워크 ID 비트 수	호스트 ID 비트 수	시작주소	끝 주소
Class A	0	8	24	0.0.0.0	127.255.255.255
Class B	10	16	16	128.0.0.0	191.255.255.255
Class C	110	24	8	192.0.0.0	223.255.255.255

- 다음 IP 주소의 클래스는?

✓ 26.48.14.2 : A

✓ 192.168.0.1 : C

A: 0 ~ 127

B: 128 ~ 191

C: 192 ~ 255

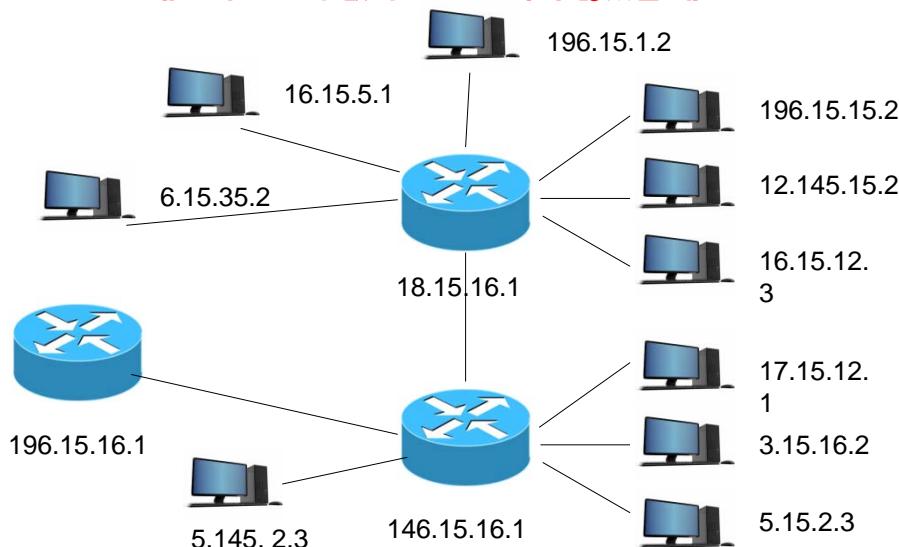
# IP 주소체계[4/8]

## □ IP 주소체계

### ▪ 네트워크 ID와 호스트 ID로 구분하는 이유

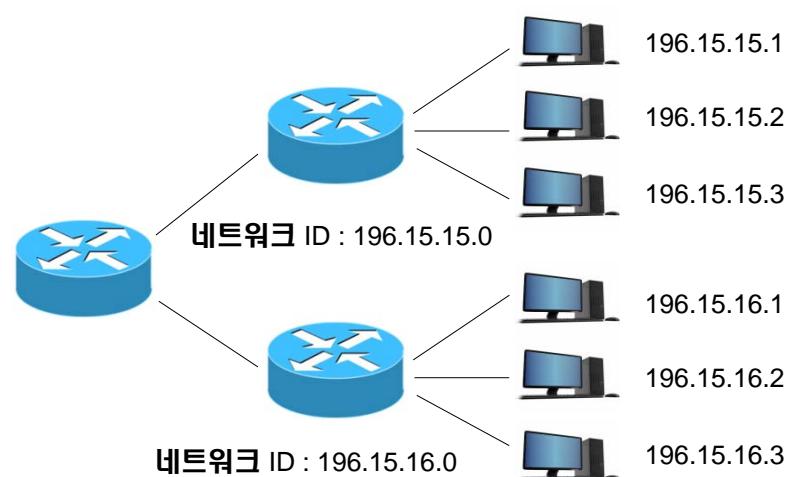
- ✓ 제한된 IP 수를 효율적으로 사용하기 위하여
- ✓ 네트워크 ID를 기반으로 그룹화하면 경로 검색시 효율 증대
  - \* 네트워크 ID는 호스트 ID 영역을 모두 0으로 하여 표시

네트워크 ID가 없어 그룹화 되지 않았을 때



목적지 주소가 196.15.15.2/24인 메시지가 도착하면,  
IP 주소가 그룹화되지 않았기 때문에 하나하나씩 모두  
찾아야 함 ➔ 과부하 발생

네트워크 ID를 이용하여 그룹화하였을 때



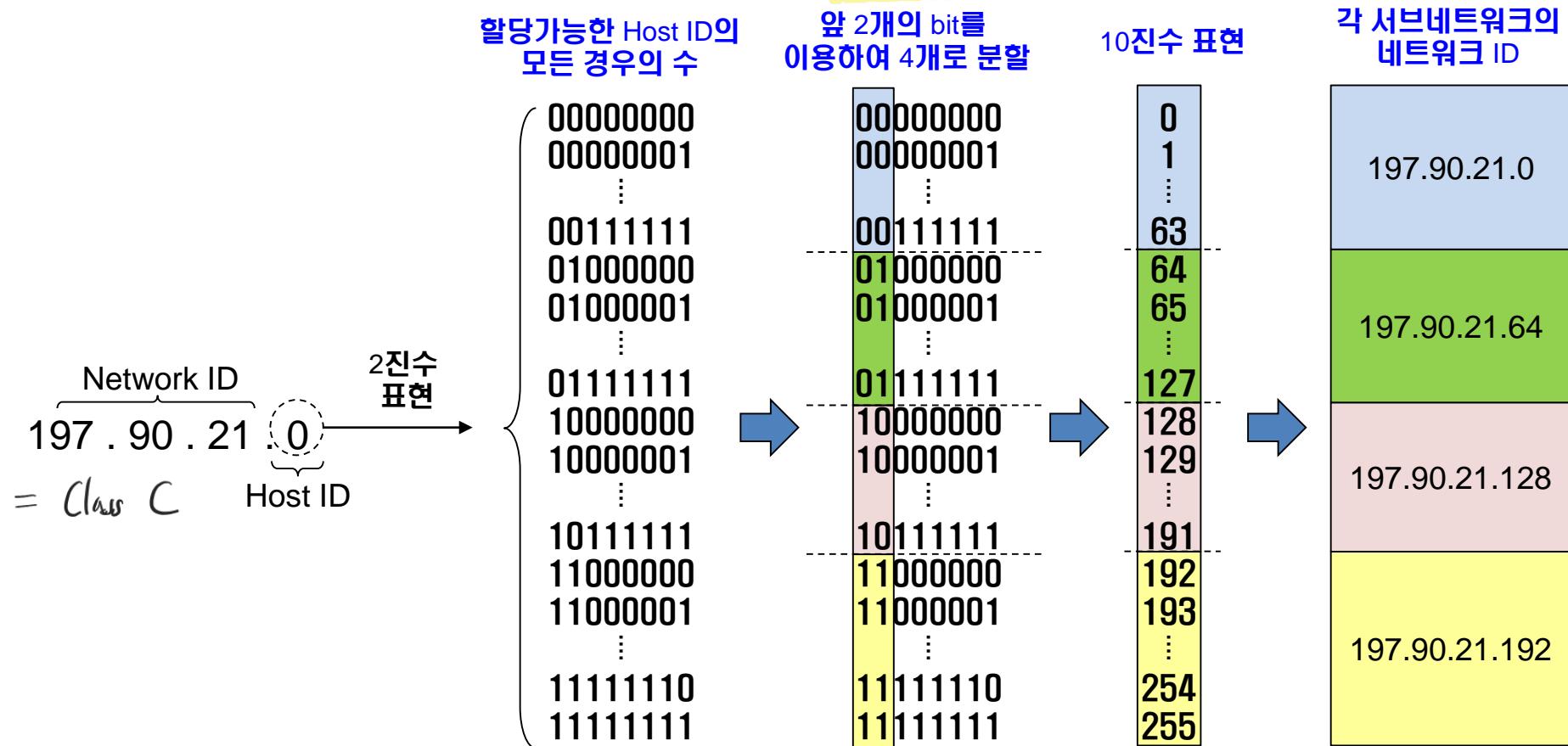
목적지 주소가 196.15.15.2/24인 메시지가 도착하면,  
라우터는 네트워크 ID 196.15.15.0을 참고하여  
해당 라우터로만 데이터 전송

# IP 주소체계[5/8]

## □ 서브넷팅(Subnetting)

- IP주소를 분할하여 2개 이상의 소규모 네트워크로 구성하는 것

예) C 클래스 네트워크를 4개의 하위 네트워크로 서브넷팅 하는 경우





# IP 주소체계[6/8]

## ■ 서브넷 마스크(Subnet Mask)

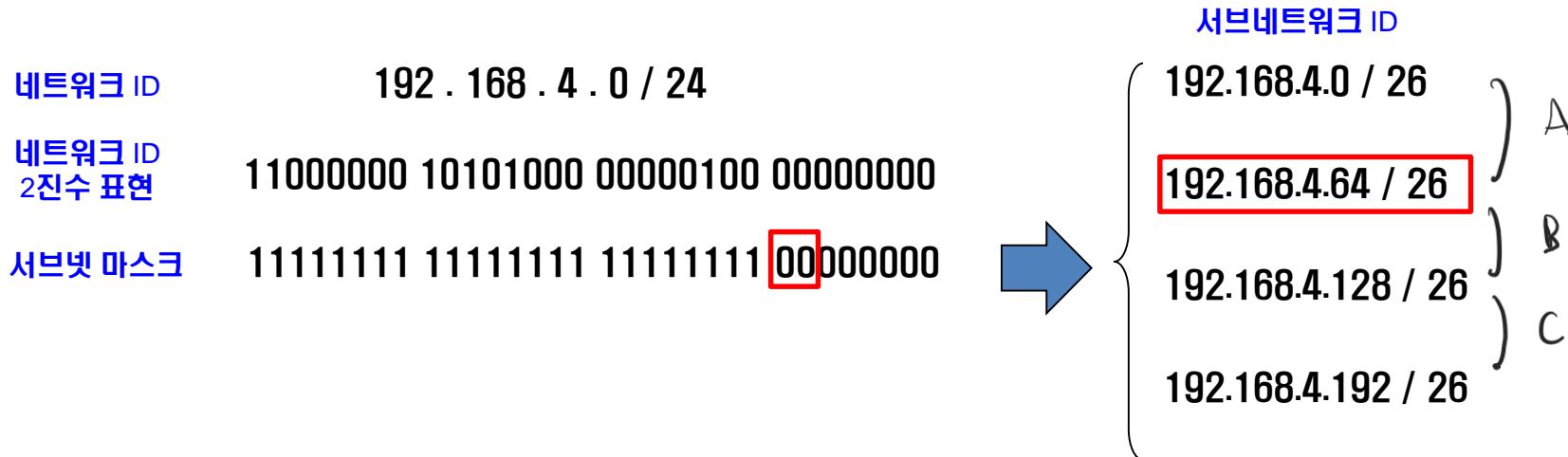
- ✓ IP 주소의 32비트 중 네트워크ID로 첫 번째 비트에서 몇 번째 비트까지 사용했는지 알려주는 것
  - \* 호스트가 로컬 서브넷에 있는지 다른 네트워크에 있는지 확인할 때 쓰임
- ✓ DDN(Dotted Decimal Notation) 또는 CIDR(Classless inter-domain Routing)로 표기
- ✓ 네트워크 ID에는 비트 1을 부여하고 호스트 ID에는 비트 0를 부여

\* A, B, C 클래스의 서브넷 마스크 예)

클래스	이진수	DDN	CIDR
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

# IP 주소체계(7/8)

예) C 클래스 네트워크를 4개의 하위 네트워크로 서브넷팅 하는 경우



호스트 수 : 192.168.4.0 / 26 → 192.168.4.1 / 26 ~ 192.168.4.62 / 26 ( $2^6 - 2 = 62$ 개)

※ IP Address 와 서브넷 마스크를 이용한 네트워크 ID 식별 방법

IP Address	192 . 168 . 4 . 67 / 26
IP Address 2진수 표현	11000000 10101000 00000100 01000011
서브넷 마스크	11111111 11111111 11111111 11000000
서브네트워크 ID	11000000 10101000 00000100 01000000

A	B	out
0	0	0
0	1	0
1	0	0
1	1	1

〈AND 연산〉

192.168.4.64 / 26

# IP 주소체계[8/8]

## □ IP version 6

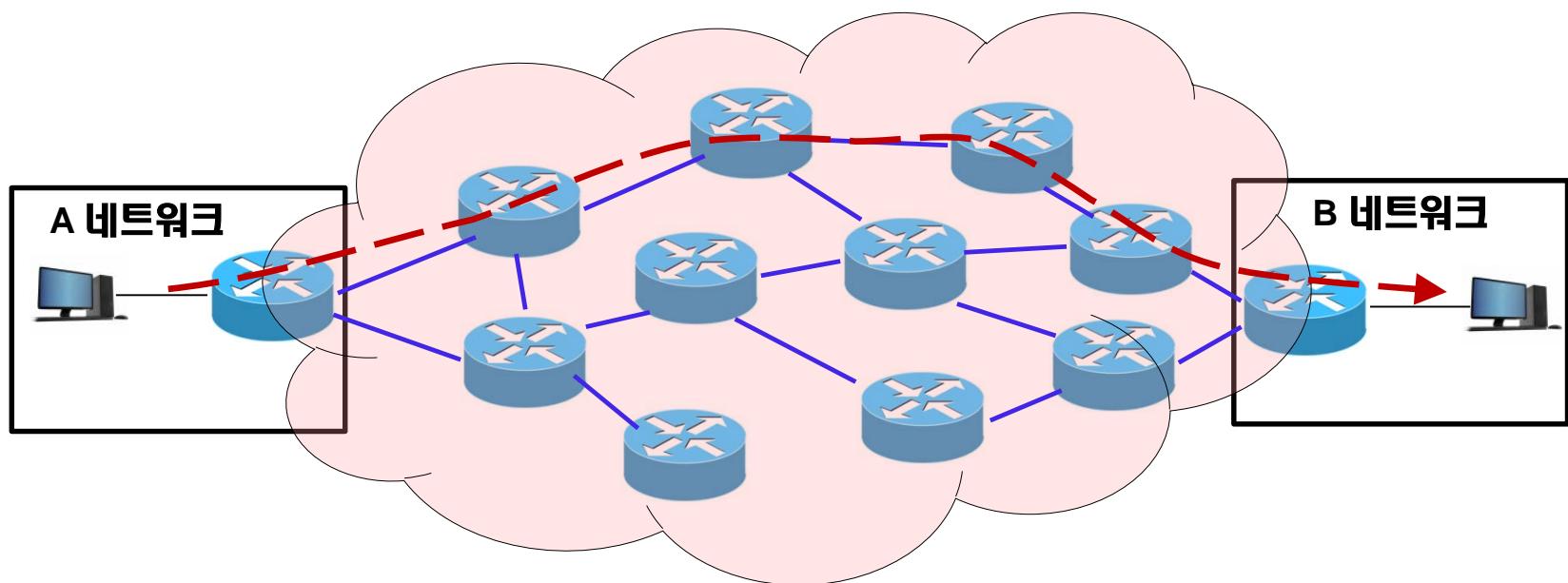
- IP 부족 문제 및 기존 IP 프로토콜의 문제 해결하기 위해 개발
  - ※ 현재는 가상 IP 등의 방법으로 IP 부족 문제를 해결하고 있음
- 32bit의 IPv4의 주소를 128bit로 확장하여 주소의 개수가 큰 폭으로 증가
  - ※ IPv4 주소의 숫자 :  $2^{32} \approx 4.29 \times 10^9$  ↪ IPv6 주소의 숫자 :  $2^{128} \approx 3.4 \times 10^{38}$
- 주소의 표현 : 4개의 16진수의 숫자들이 하나의 그룹을 형성하고, 8개의 그룹으로 하나의 주소를 표현
- 기존 IPv4와의 호환성을 최대로 하는 방향으로 설계



# 라우팅(1/5)

## □ 라우팅이란? � 노드

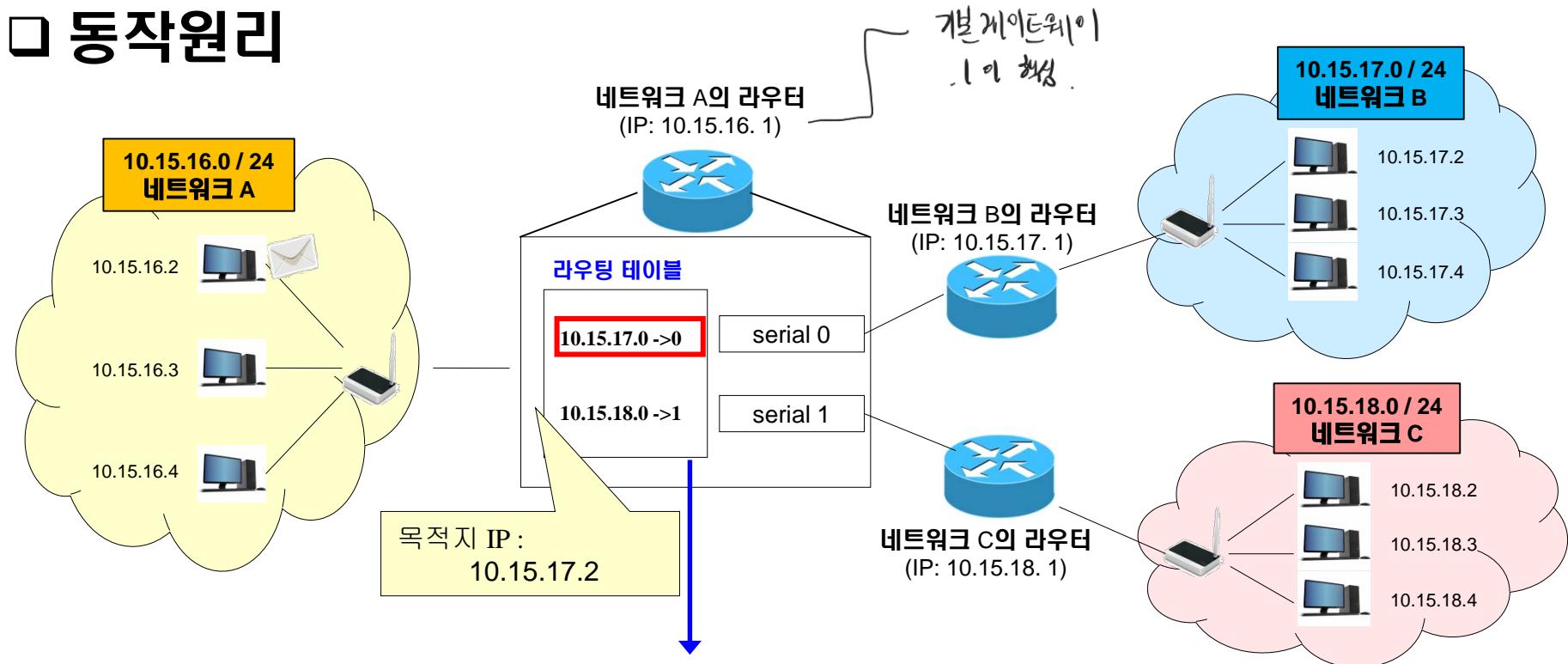
- 데이터를 가장 빠르고 효율적으로 전송하기 위해 **전송 경로를 판단하는 과정**
  - \* 라우터 : 가장 적절한 통신경로를 지정하여 중계해주는 장치
- 정적라우팅과 동적라우팅으로 구분 (경로정보 등록 주체에 따라)



# 라우팅(2/5)

□ 정적라우팅 : 관리자가 입력한 경로(라우팅테이블) 대로 정보를 전달

□ 동작원리



\* 라우팅 테이블 요소 : 목적지 네트워크 ID, 서브넷마스크, 인터페이스 주소 또는 바로 다음 라우터 주소

10. 15. 17. 0

255.255.255.0

serial0 또는 10.15.17.1

10. 15. 18. 0

255.255.255.0

serial1 또는 10.15.18.1

# 라우팅(3/5)

- 동적라우팅 : 라우터끼리 라우팅테이블을 교환하여 경로를 찾고 그 경로대로 정보를 전달

가장 최소의 라우터를 거치는 경로 찾음.  
각 Edge에서 cost를 계산하여  
최소의 값을 찾음.

- 동적라우팅 프로토콜의 종류 : RIP, OSPF, BGP, IGRP 등

\* RIP : Routing Information Protocol  
BGP : Border Gateway Protocol

OSPF : Open Shortest Path First  
IGRP : Interior Gateway Routing Protocol

- 경로결정의 요소

- 라우터를 몇 개를 거치는가?
- 데이터를 목적지까지 보내는데 시간이 얼마나 걸리는가?
- 한번에 얼마나 많은 데이터를 보낼 수 있는가?
- 어떤 경로를 안정적으로 이용할 수 있는가?
- 특정 시점에 트래픽이 과하게 사용되지 않는가?



# 라우팅(4/5)

## □ RIP(Routing Information Protocol, 경로 정보 프로토콜)

- 목적지로 가는 여러 경로 중 경유 라우터의 수가 가장 적은 경로 선택
  - ✓ 경유하는 라우터의 수를 홉(hop) 수라고 함
  - ✓ 최대 15개의 홉 수를 지원
- 라우팅테이블 수정 : 인접 라우터와 교환하여 자신의 테이블 수정

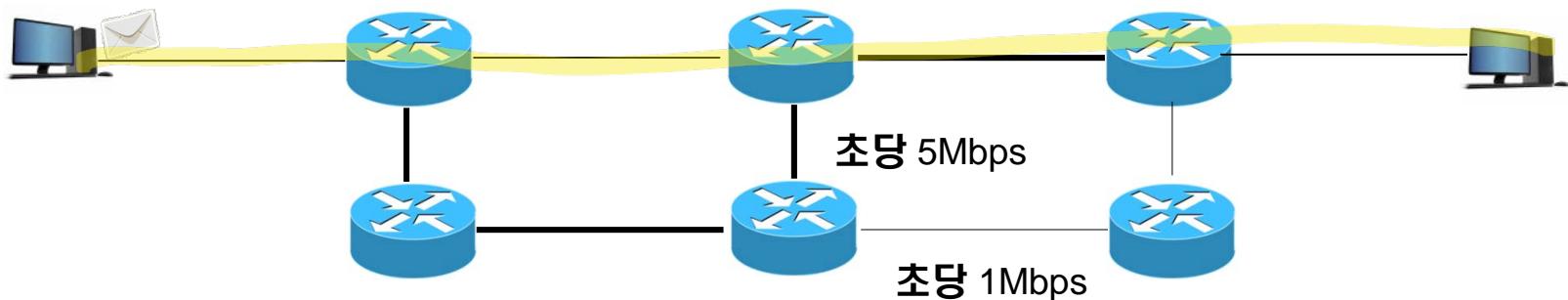
## □ OSPF(Open Shortest Path First, 개방형 최단경로 프로토콜)

- 실시간 네트워크 상태에 따라 최소 비용 경로 선택
  - ✓ 각 라우터의 데이터처리량, 왕복시간, 신뢰성 등을 기반으로 비용 산정
- 라우팅테이블 수정 : 지역 내 모든 라우터의 경로비용을 계산하여 수정

# 라우팅(5/5)

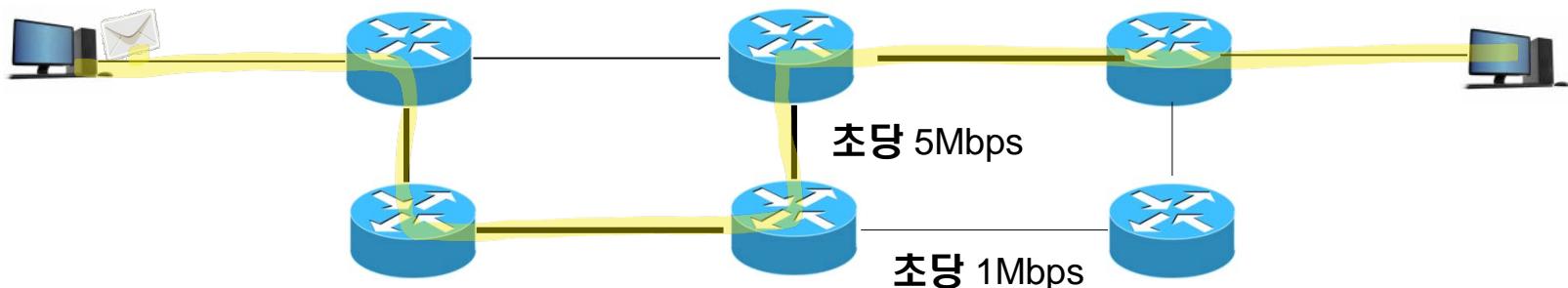
## □ RIP (Routing Information Protocol) 동작 예

- 가장 오래된 동적 라우팅 프로토콜, 라우터 수가 가장 적은 경로를 선택



## □ OSPF (Open Shortest Path First) 동작 예

- 링크상태에 따라 가장 적합한 경로를 선택하여 효율성을 향상시킨 기술





# 전송 계층

## □ 전송계층

- 단말장비에 있는 논리적 주체인 포트의 연결을 제공
  - ✓ 응용계층과 연결해 주기 위한 통로
- 대표적인 기능은 흐름제어, 오류제어, 혼잡제어를 수행
  - ✓ 흐름제어 : 수신자가 받을 수 있는 상태인지를 확인하면서 송신자의 데이터 송출속도를 조절하는 것
  - ✓ 오류제어 : 응용계층에 오류 없는 데이터를 보내기 위한 오류 확인 및 복원
  - ✓ 혼잡제어 : 네트워크의 혼잡상태를 판단하여 전송할 데이터를 네트워크에 내보낼지 말지를 결정하는 과정
- 대표적 프로토콜 : TCP (연결형), UDP (비연결형)  
※ TCP : Transmission Control Protocol / UDP : User Datagram Protocol

# TCP(1/9)

## □ TCP (Transmission Control Protocol)

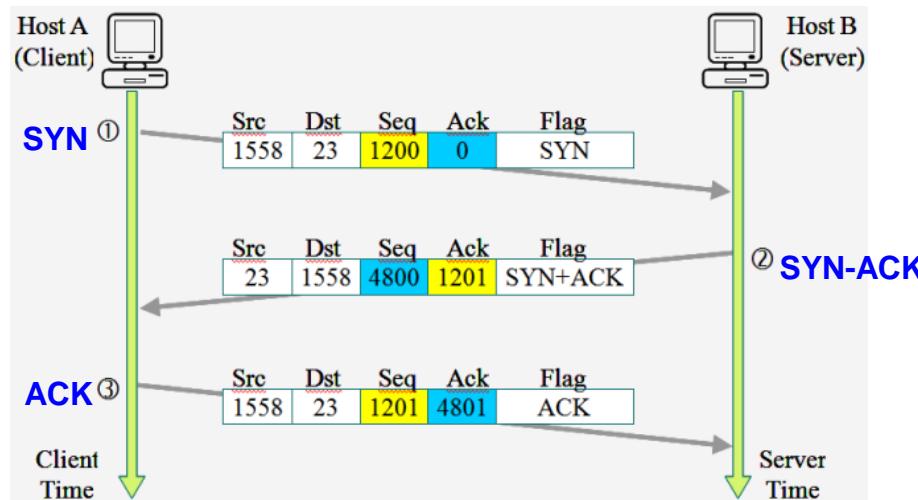
- 전달계층의 대표적인 프로토콜로서, **연결형 서비스**를 지원
- **응용계층**으로부터 전송 요청된 데이터를 분할하여 **세그먼트(Segment)**라고 하는 **블록**으로 나누어 하위 계층인 IP 계층에 전달하여 통신요청
- “**연결형**”이란 송신 및 수신 포트가 **논리적 연결을 수립하면, 연결을 끊지 않고 스트리밍 방식으로 연속적으로 데이터를 보내는 방식**을 의미
- 3단계의 과정으로 송수신 단말이 연결됨



# TCP(2/9)

## □ 접속구축

- Three-Way Handshake를 통하여 송신과 수신단말 사이의 연결이 수립
- Three-Way Handshake 과정
  - ① SYN : 송신단말이 SYN 패킷(수신노드에게 수신을 요청하는 패킷) 전송.  
이때 임의의 순서번호 A 지정
  - ② SYN-ACK : 수신노드의 SYN에 대한 응답으로서 SYN-ACK 패킷을 송신노드에게 전송.  
SYN-ACK에는 ①의 SYN에 대한 응답이라는 것을 표시하기 위해  
SYN 패킷의 Seq No. A에 1을 더하여 전송, 새로운 임의의 Seq No. B 지정



# TCP(3/9)

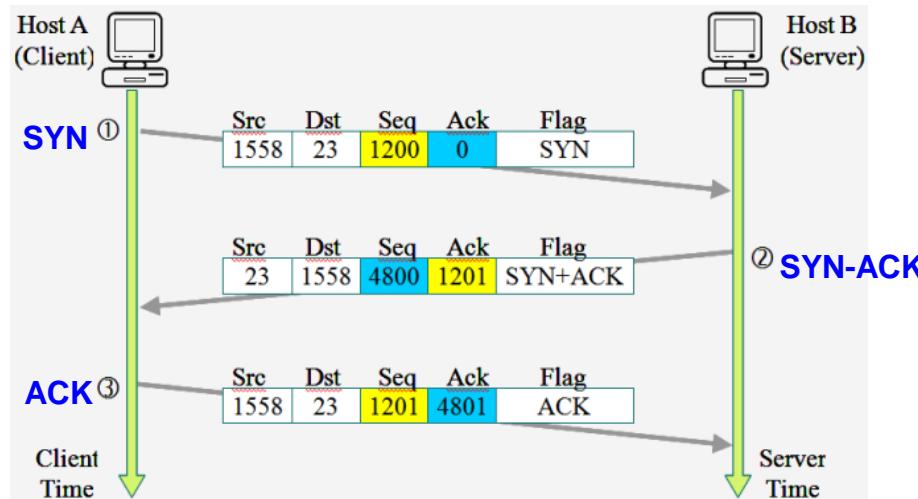
## □ 접속구축

### ▪ Three-Way Handshake 과정(계속)

③ ACK : 송신노드는 ACK 패킷을 목적지에 전송하여 접속을 구축. Seq No는 A+1, SYN-ACK에 대한 응답표시를 위해 Ack에 B+1을 지정하여 전송

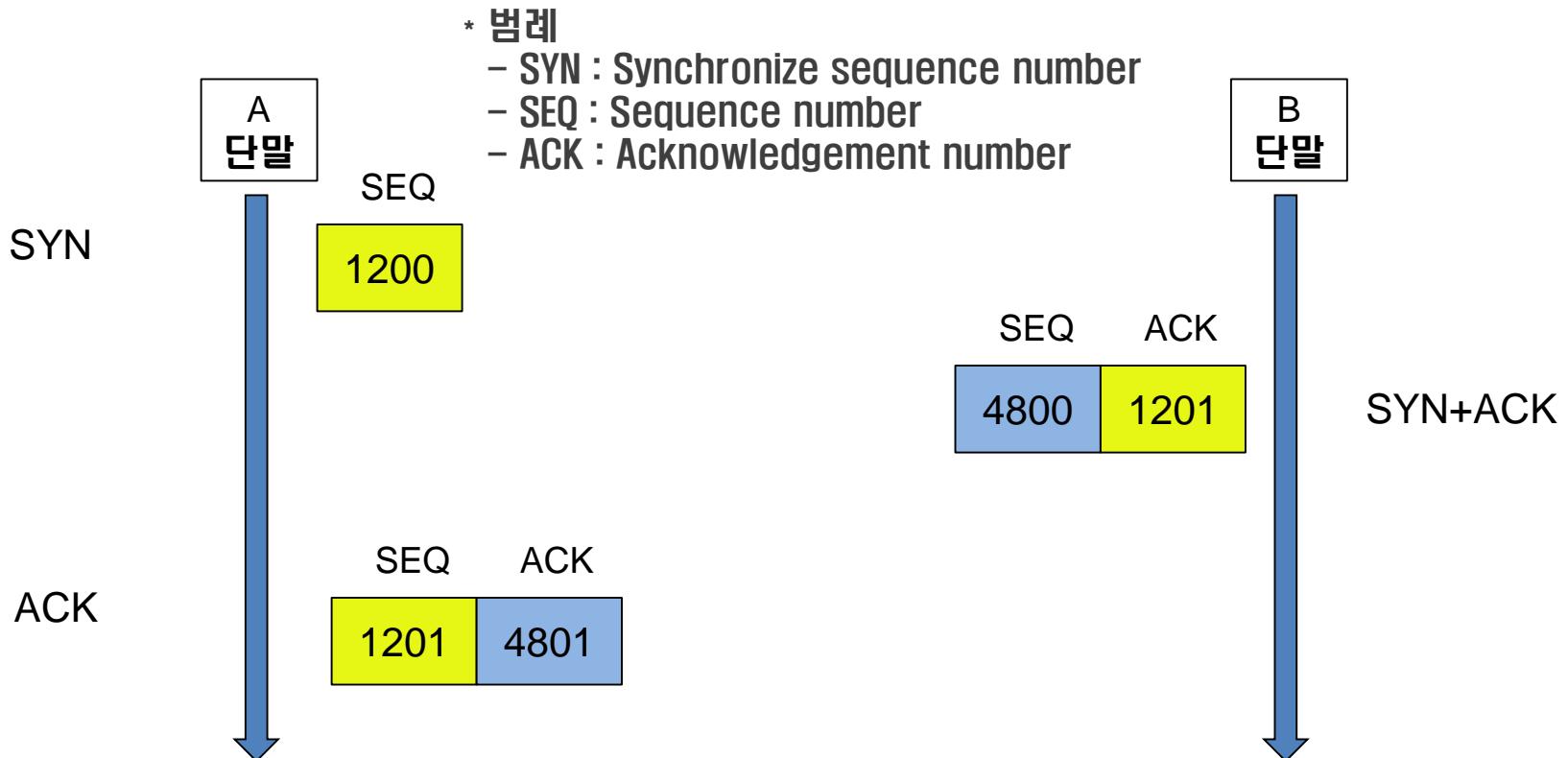
①~② 과정을 통해 송신노드가 수신노드에게 접속되었다는 사실을 알 수 있고,

②~③ 과정을 통해 수신노드는 송신노드가 자신에게 접속하였다는 사실을 알 수 있어, 전이중통신을 위한 사전 준비단계라고 할 수 있음



# TCP(4/9)

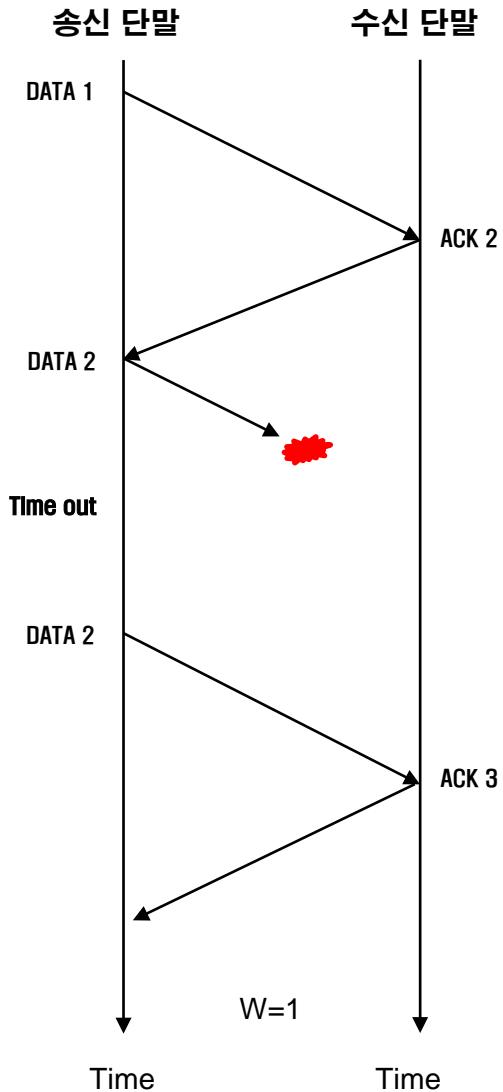
## □ 접속구축(예시)



# TCP(5/9)

## □ 접속유지

- 연결이 성립되면 송신단말과 수신단말은 연속적으로 데이터를 효율적으로 전송
  - ☞ 오류제어, 흐름제어, 혼잡제어
- 오류제어
  - ✓ ACK – retransmission 방법
    - 전송과정 중에 오류가 발생하여 데이터가 분실 되었을 때, 해당 데이터를 재전송하여 오류를 복구하는 방법





# TCP(6/9)

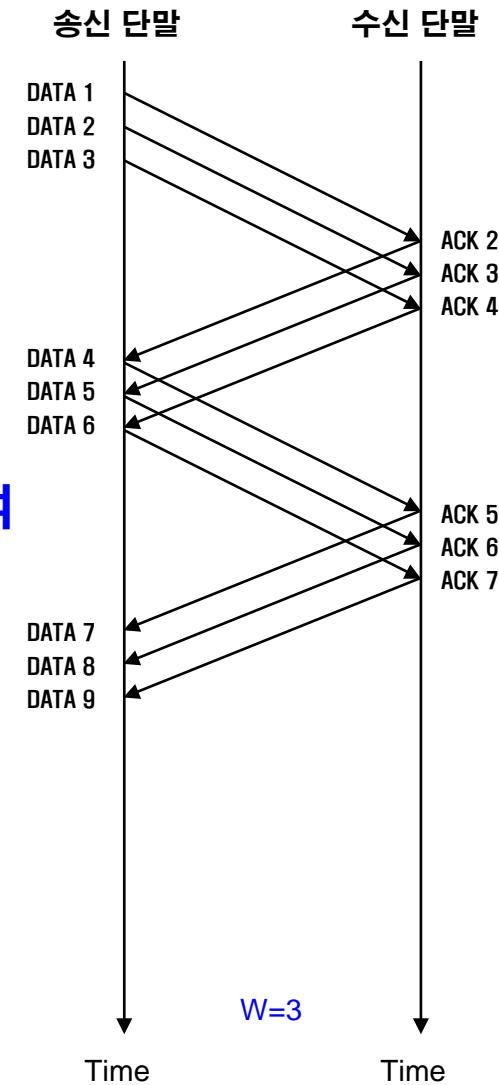
## □ 접속유지

### ▪ 흐름제어

- ✓ 수신자는 ACK 패킷 안에 자신이 가능한 **버퍼 사이즈** (슬라이딩 윈도우의 크기)를 기입하여 송신자로 하여금 데이터 발생률을 조정하게 함
- ✓ 수신자의 수신 능력에 따라 **데이터 발생률을 조정하여 전송**

### ▪ 슬라이딩 윈도우 (Sliding Window)

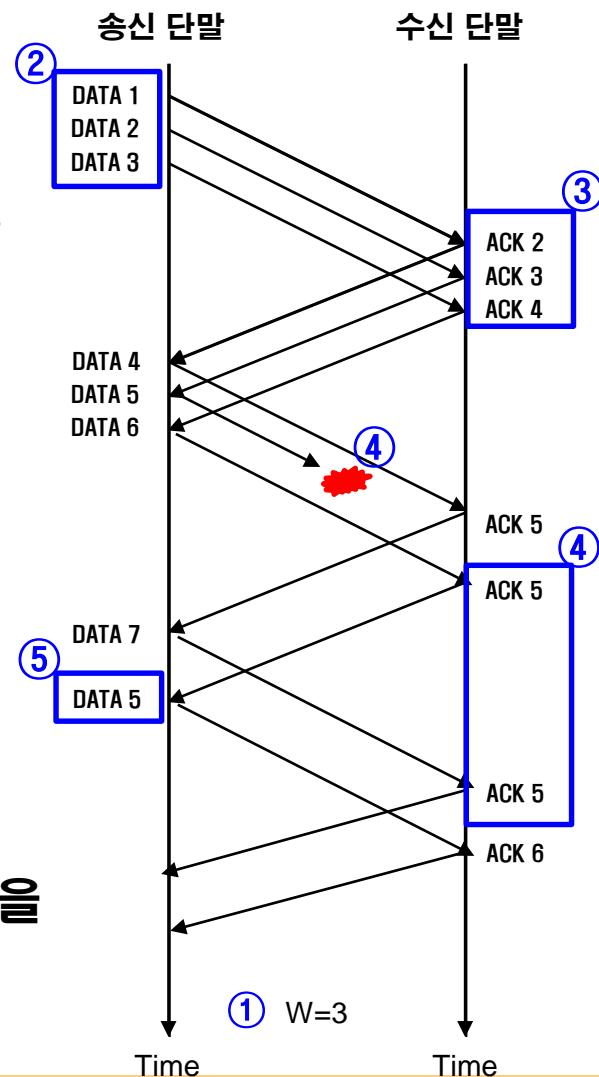
- ✓ 슬라이딩 윈도우( $W$ )라고 하는 크기를 부여
- ✓  $W$  크기에 해당하는 데이터들을 연속적으로 전송



# TCP(7/9)

## □ 접속유지

- 오류제어의 예 (with 슬라이딩 윈도우)
  - ① 송신단말의 슬라이딩 윈도우의 크기는 3으로 가정
  - ② 접속이 구축되면 송신단말은 3개의 세그먼트 (DATA 1 ~3)를 전송
  - ③ 이 세그먼트를 수신한 수신단말은 각 세그먼트를 이상 없이 수신한 경우 각 세그먼트의 순서번호에 1을 추가하여 ACK 패킷 전송
  - ④ DATA 5가 전송과정에서 오류가 났기 때문에, 수신단말은 DATA 6, DATA 7를 수신하였을 경우에도 ACK5를 전송하고 DATA 6, DATA 7 버림
  - ⑤ 두 번째 ACK5를 수신한 송신 단말은 오류가 난 것을 확인하고 재전송하여 오류를 복원



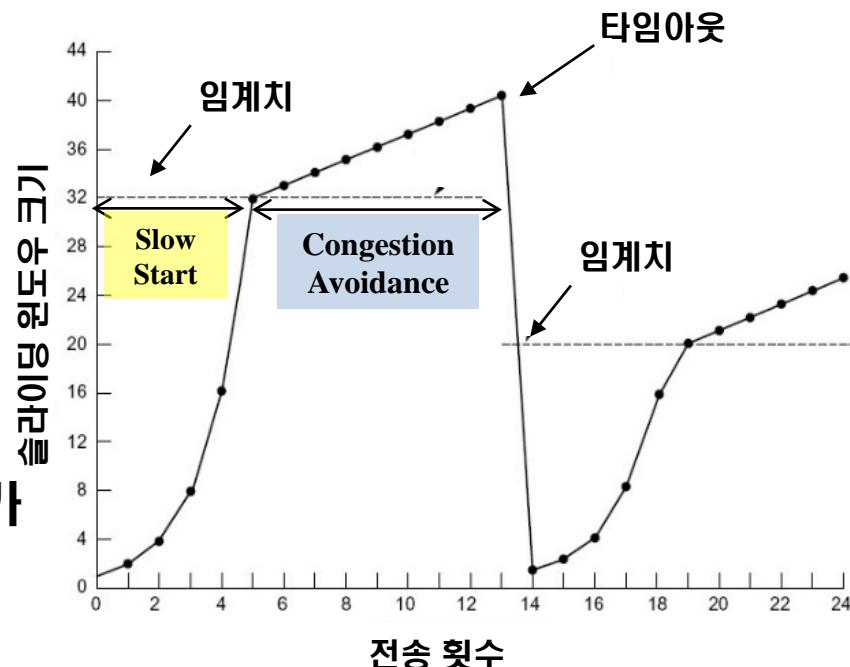
# TCP(8/9)

## □ 접속유지

- 혼잡제어 (Congestion Control) 
  - ✓ 네트워크 상태를 고려하여 데이터의 전송 속도를 조절하는 기법
  - ✓ ACK 패킷의 수신되는 정도를 통해서 네트워크의 혼잡상태를 파악 후 슬라이딩 윈도우  $W$ 의 크기를 조절
  - ✓ 혼잡 제어의 대표적인 알고리즘은 Slow-Start, Congestion Avoidance가 있음

### ※ 자신의 슬라이딩 윈도우 $W$ 크기 결정

- 혼잡제어의 윈도우 사이즈 (Flow control window)
- 혼잡제어의 윈도우 사이즈 (Congestion control window)
- 두 개 값 중에 최소값으로 슬라이딩 윈도우  $W$  크기 결정

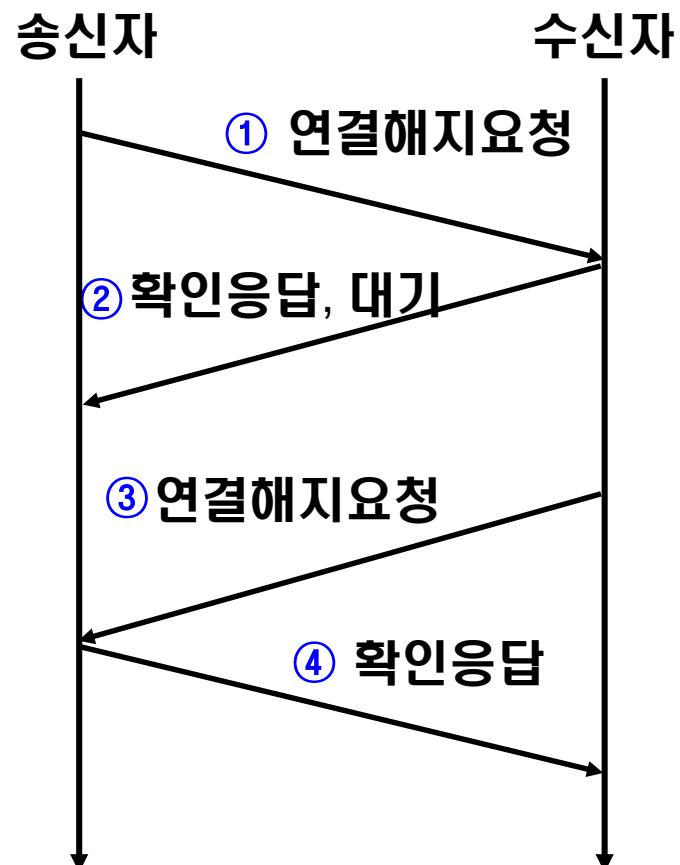


최소값  $\rightarrow W$  크기 결정.

# TCP(9/9)

## □ 접속해제

- 접속해제는 Four-way handshake 과정을 통해 이루어짐
    - ① 송신 단말이 연결을 종료하겠다는 메시지 전송
    - ② 수신 단말은 확인메시지를 보내고 자신의 통신이 끝날때까지 기다림
    - ③ 수신 단말이 통신이 끝나면 연결이 종료되었다고 송신 단말에게 메시지 전송
    - ④ 송신 단말은 확인했다는 메시지를 전송
- ※ 접속구축과 유사한 과정



# UDP

## □ UDP (User Datagram Protocol)

- TCP에 비해 훨씬 간단한 헤더 구조를 갖는 전송계층 프로토콜  
※ 상대적으로 신뢰도가 낮으며, 비연결형 서비스 제공
- 방송(broadcasting)과 같이 많은 사람에게 동영상 정보를 전송하면서 데이터 하나하나마다 오류 발생 여부를 확인하는 것이 비효율적이고 불가능할 경우에 사용 \* 아프리카 TV, 시간서버(Network Time Protocol)
- 오류제어, 흐름제어, 혼잡제어 기능 미실시

항목	프로토콜	TCP	UDP
서비스	연결형 서비스 (connection-oriented)	비연결형 서비스 (connectionless)	
수신순서	송신순서와 동일	송신순서와 다를 수 있음	
오류제어 및 흐름제어	있음	거의 없음	
응용 소프트웨어	웹 브라우징, 파일 전송	제어신호, DNS, 동영상 전송 등	

<TCP와 UDP 비교>



# 무선 네트워크(1/5)

## □ 네트워크

- 전송로에 따라 유선네트워크, 무선네트워크로 구분
- 네트워크의 크기에 따라 LAN, MAN, WAN

※ LAN : Local Area Network   MAN : Metropolitan Area Network   WAN : Wide Area Network

## □ 무선 네트워크

- 전파를 활용하여 통신을 하는 네트워크를 통칭
- 이를 해결하기 위하여 교환노드들 간의 링크가 무선으로 연결되거나, 교환노드들이 전송매체를 무선으로 공유하는 무선 네트워크를 사용함
- 무선 네트워크의 종류 : Ad-Hoc, WMN, IoT

※ WMN : Wireless Mesh Network

IoT : Internet of Things

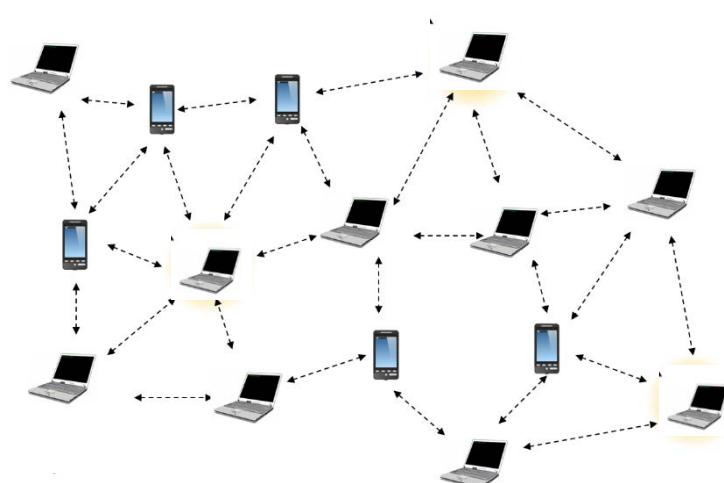


# 무선 네트워크(2/5)

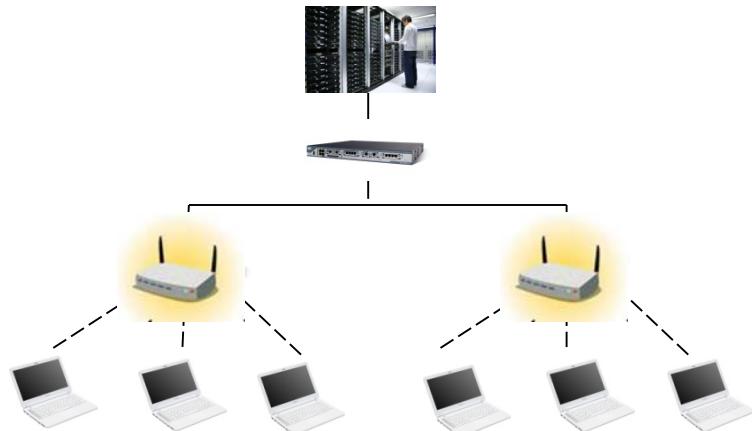
## □ Ad-Hoc Network

(Ad hoc : 라틴어로 'for this')

- 별도의 기반시설 없이 이동통신장비만으로 구성 가능한 네트워크
  - \* 라우터, Access Point 등 불필요
- 노드와 단말의 구분이 없으며, 단말장비도 중계기 역할을 수행함
  - \* 통신거리의 제약을 극복할 수 있음
- 노드, 단말장비 모두 무선으로 연결되어 이동간 통신 보장



<Ad-Hoc 네트워크>



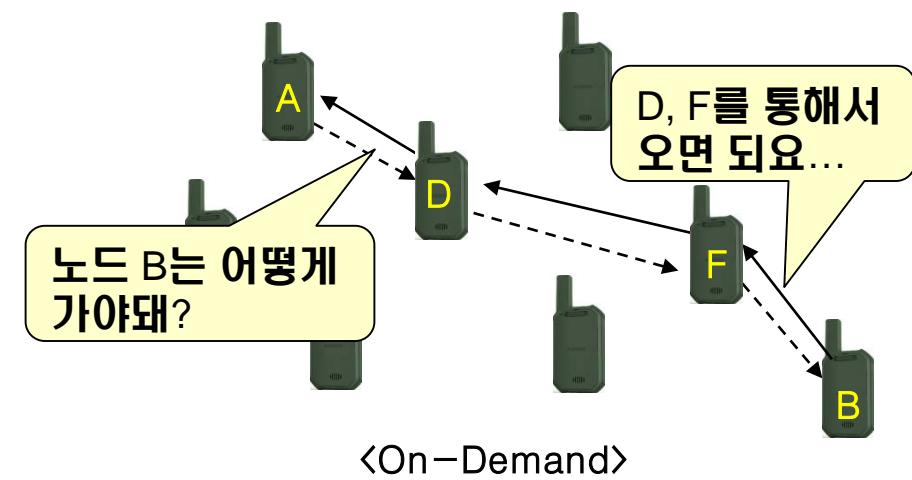
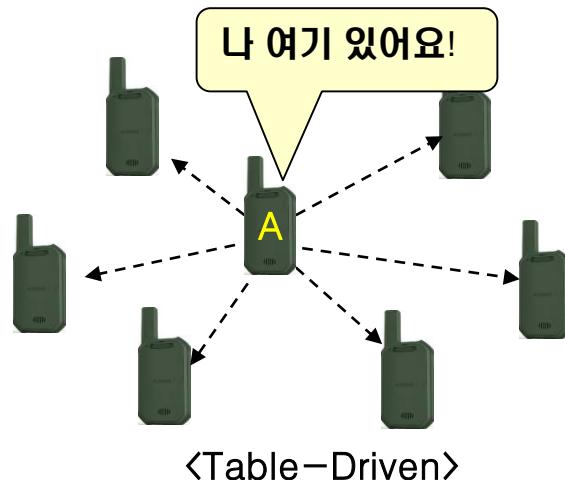
<Infrastructure 네트워크>

# 무선 네트워크(3/5)

## □ Ad-Hoc Network

### ▪ 라우팅 기술

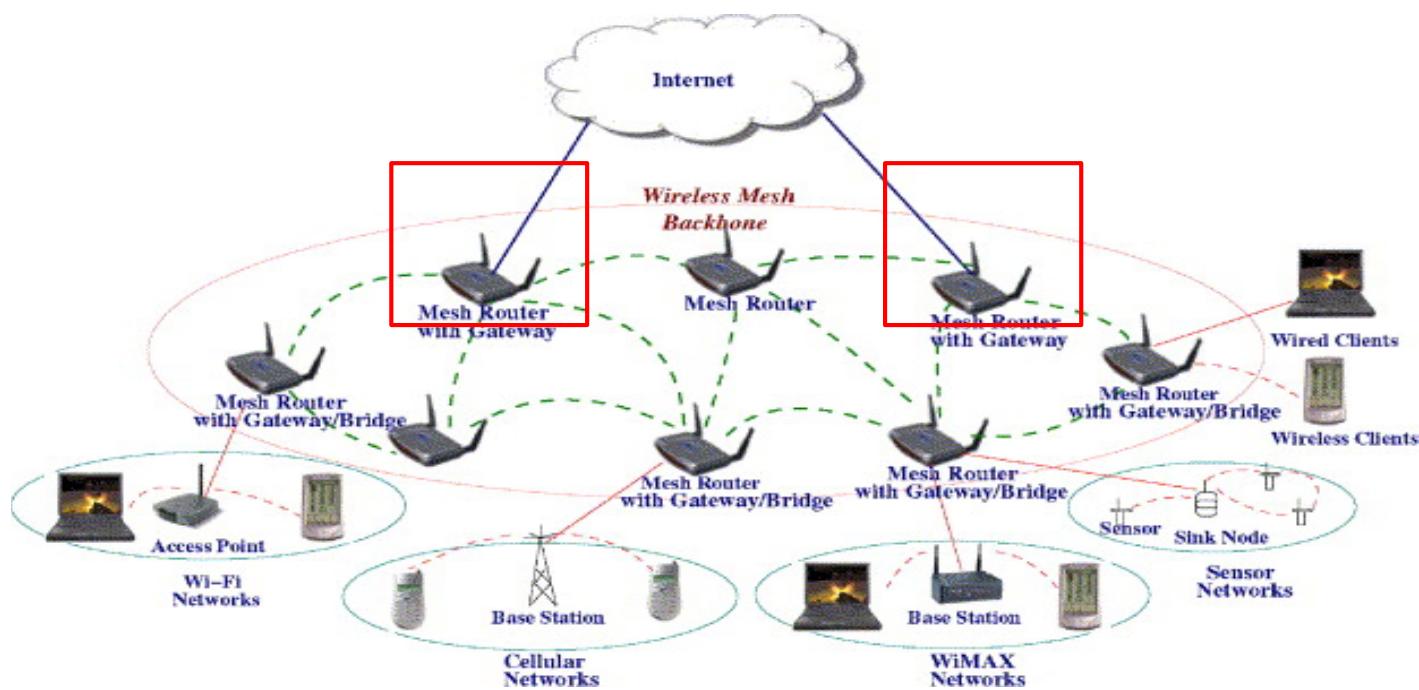
- ✓ Table-Driven 방식 : 주기적으로 또는 네트워크 토폴로지가 변화할 때마다 라우팅 정보를 브로드캐스팅하여 모든 노드가 최신 라우팅정보 획득
- ✓ On-Demand 방식 : 정보를 전송하고자 하는 노드가 목적지 노드의 경로가 필요할 때 요청하여 라우팅정보를 습득하는 방식
  - \* 네트워크에서 라우팅 오버헤드는 On-Demand 방식이 적음



# 무선 네트워크(4/5)

## □ WMN (Wireless Mesh Network)

- 대표 AP만 유선으로 인터넷과 연결되고 다른 무선통신 라우터들이 메쉬 노드가 되어 모든 구간을 무선으로 연결해가는 방식의 네트워크
  - \* 메쉬라우터는 이동성이 없으며, 메쉬노드는 이동성 유무와 관계 없음



# 무선 네트워크(5/5)

## □ WSN (Wireless Sensor Network)

- 무선 통신기능을 갖춘 센서 노드들이 온도, 소리, 압력 등과 같은 현상을 측정하여 메인 노드에게 전송하는 네트워크
- 메인노드는 Sink Node로 불리며, 데이터를 종합, 해석하고 인터넷과 연결
- 응용 : 적 침입 알림, 공기오염감지, 산불감지, 산사태감지 시스템 등
- 기술 : ZigBee, 6LoWPAN, IoT 등

