

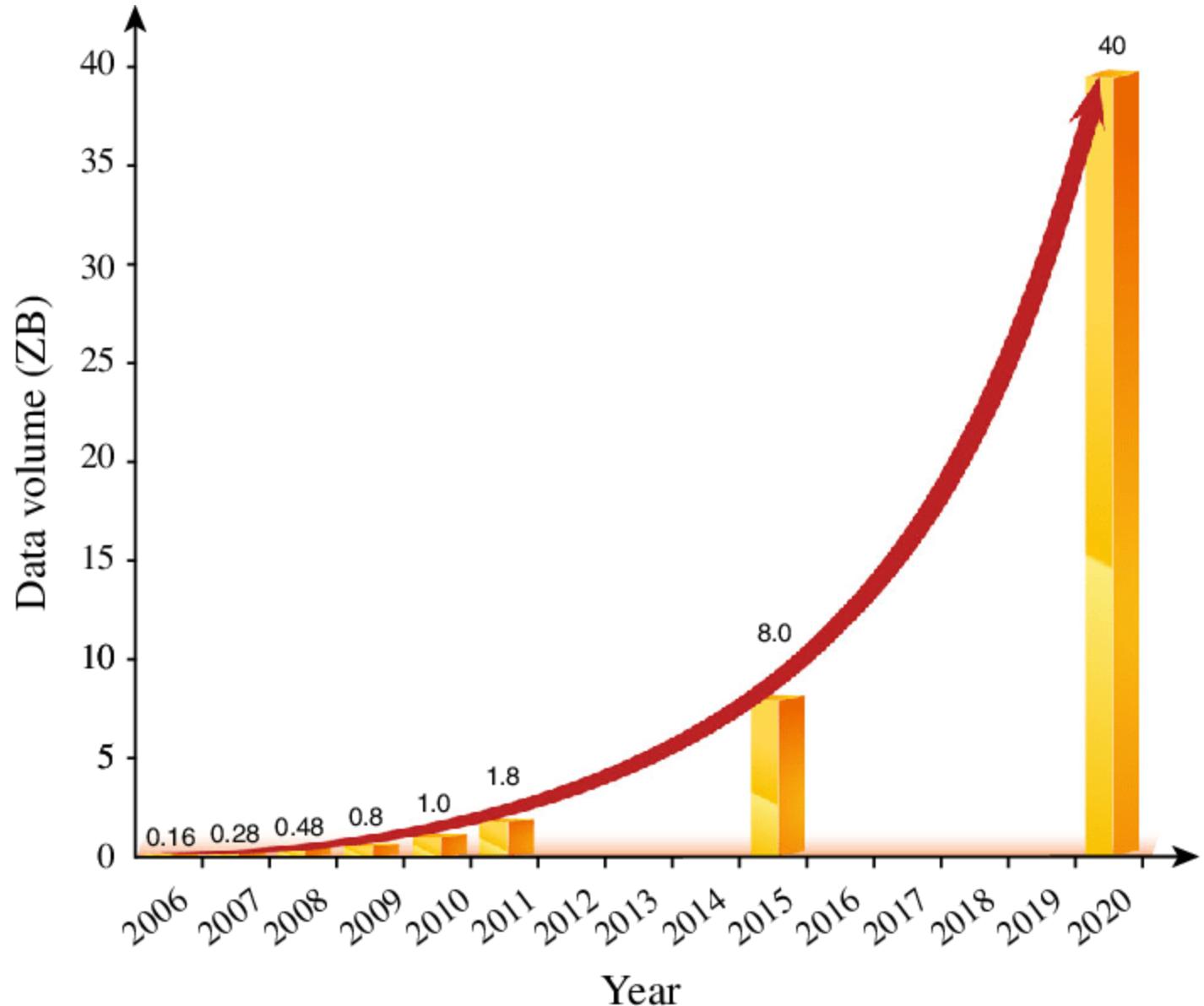


# 컴퓨터 네트워크와 보안

2022. 4. 16, 17, 23 (3일)

# 컴퓨터 네트워크 수업 일정

일정	학습 내용	상세 내용
1일차	과정 소개	네트워크 분석 과정 소개
		네트워크의 HW와 SW 구성
	점심시간	
		MAC Address; 가까이 있는 컴퓨터의 데이터를 주고 받기
		IP Address: 컴퓨터간의 IP주소를 사용해 데이터를 주고 받기
		라우팅 테이블 및 전송 과정: 멀리 있는 컴퓨터의 데이터 주고 받기
2일차		컴퓨터의 프로그램의 데이터를 주고 받기
		<b>실습 프로그램: wireshark virtualbox</b>
		TCP와 UDP
		NAT와 포트 포워딩
	점심시간	
		HTTP: www(웹)를 이용한 데이터를 주고 받기
3일차		URL, URI: www(웹)를 이용한 데이터를 주고 받기
		<b>실습 프로그램: virtualbox burpsuite</b>
		방화벽
		IDS/IPS
	점심시간	
		ACL
		Router
		<b>실습 프로그램: wireshark snort</b>



# 컴퓨터 네트워크 수업 목차

1. 네트워크란?
2. 네트워크 모델
3. 가까이 있는 컴퓨터의 통신
4. 컴퓨터의 IP주소를 이용한 데이터 통신
5. ARP 프로토콜
6. 멀리 있는 컴퓨터의 데이터 통신
7. UDP와 TCP
8. NAT와 PORT 포워딩
9. HTTP

# 네트워크 Hardware

## 허브



허브는 OSI 7 Layer 1계층 장비이며, 컴퓨터의 LAN 카드는 케이블을 통하여 허브의 포트와 연결됩니다. 허브를 통해 연결된 컴퓨터들은 서로 데이터 통신이 가능한데, 이때 A컴퓨터가 B컴퓨터에게 데이터를 전송할 때, 허브는 B뿐만 아니라 허브와 연결되어 있는 C,D 컴퓨터들에게도 데이터를 전송합니다. C,D 컴퓨터는 LAN 카드에서 해당 데이터가 자신에게 온 것이 아닌 것을 확인하고 폐기하는데 이런 방법을 너무 비효율 적이며 A가 통신을 하고 있을 때 B,C,D는 통신을 할 수 없으면 만약 한다면 충돌이 일어날 것입니다. 이렇게 충돌하는 영역을 collision domain이라고 하며 이런 이유로 허브의 통신을 1차선 통신이라고 표현하기도 합니다.

# 네트워크 Hardware

## 스위치



스위치(switch)는 수신한 데이터를 연결된 모든 컴퓨터에게 전송하는 허브와는 달리 정확한 목적지 컴퓨터에게만 데이터를 전송합니다. 이렇게 스위치가 정확히 목적지 컴퓨터를 식별할 수 있는 이유는 자신에게 연결된 디바이스들의 IP와 MAC 주소를 테이블로 가지고 있기 때문입니다. 따라서 데이터가 자신에게 오면 그것의 목적지가 어디인지를 파악하여 해당 컴퓨터에게 데이터를 보내줍니다.

OSI 7 layer에서 스위치는 어떤 주소를 가지고 스위칭을 하는가에 따라 L2, L3, L4, L7으로 나눌 수 있으며 일반적으로 높은 계층을 다루는 스위치일 수록 비싸며, 하위 계층의 스위치 기능을 할 수 있습니다.

# 네트워크 Hardware

# 라우터



라우터(Router)는 여러개의 독립된 네트워크를 연결, 분할, 구분 시켜주는 역할을 합니다. 같은 네트워크에서는 스위치를 통하여 다른 컴퓨터로 데이터를 전송할 수가 있습니다. 하지만 다른 네트워크에 존재하는 컴퓨터가 서로 데이터를 주고받기 위해서는 반드시 라우터를 거쳐야만 합니다.

사실상 라우터는 L3 스위치입니다. 패킷이 전송될 때 여러 라우터를 거치게 되는데 각각의 라우터는 현재 자신의 상태를 공유합니다. 인접한 라우터의 상태를 알고 있어야 그곳으로 패킷을 보낼지 다른 곳으로 보낼지 결정할 수 있기 때문입니다. 여기서 다른 라우터와 어떤 프로토콜(라우팅 프로토콜, 스패닝 트리 프로토콜)을 통해 자신의 상태를 공유하고 있는 L3 스위치를 라우터라고 합니다. 즉 라우팅 프로토콜을 지원하는 L3 스위치가 라우터입니다.

# 네트워크란 무엇인가?

네트워크란  
무엇인가?

네트워크의  
분류

네트워크의  
통신 방식

네트워크  
프로토콜

실습

네트워크란?  
인터넷이란?

크기에 따른 분류  
연결 형태에 따른 분류

네트워크에서  
데이터는  
어떻게 주고받는가?

프로토콜이란?  
여러가지 프로토콜

Wireshark 설  
프로토콜 직접 보기

# 네트워크란 무엇인가?

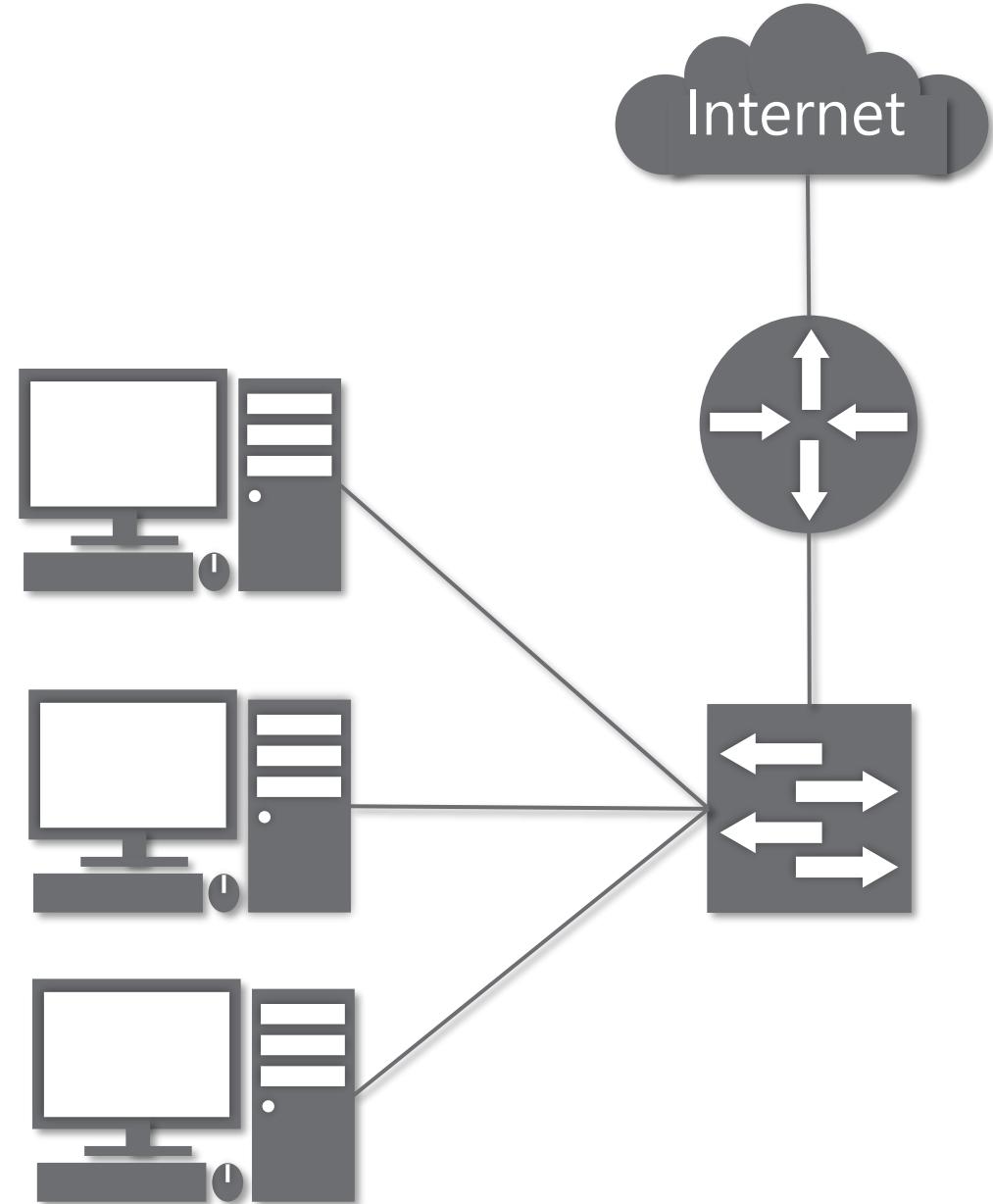
네트워크란?

노드들이 데이터를 공유할 수 있게 하는 디지털 전기통신망의 하나이다.

즉, 분산되어 있는 컴퓨터를 통신망으로 연결한 것을 말한다.

네트워크에서 여러 장치들은 노드 간 연결을 사용하여 서로에게 데이터를 교환한다.

\*노드 : 네트워크에 속한 컴퓨터 또는 통신 장비를 뜻하는 말



# 네트워크란 무엇인가?

인터넷이란?

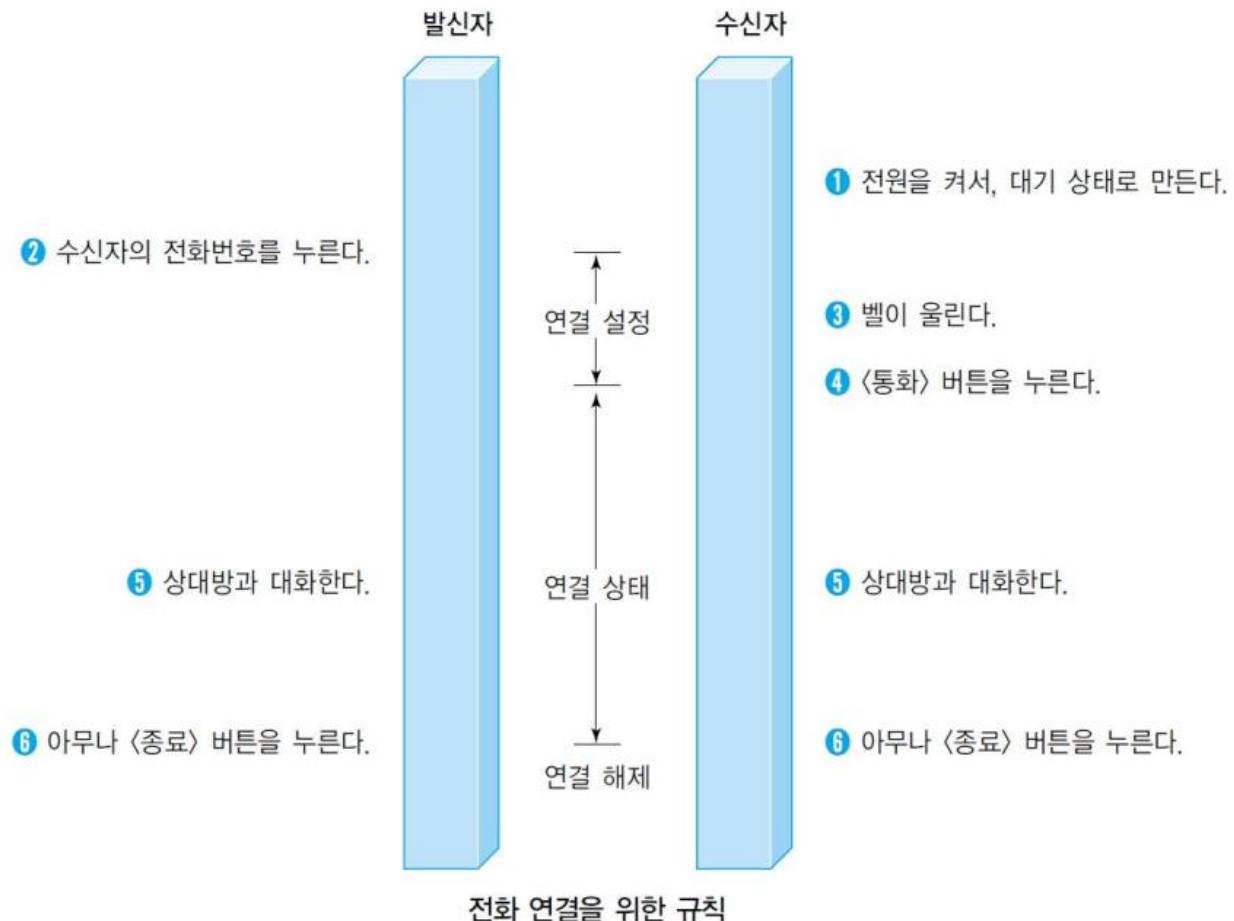
문서, 그림 영상과 같은 여러가지 데이터를  
**공유**하도록 구성된 세상에서 가장 큰  
전세계를 연결하는 **네트워크**

흔히 www를 인터넷으로 착각하는 경우가  
많은데 www는 인터넷을 통해 웹과 관련된  
데이터를 공유하는 것



# 프로토콜

통신회선을 이용하여 컴퓨터간에 데이터를 주고 받는 상호약속



# 네트워크의 분류

# 네트워크의 분류

크기에 따른 분류

〃

Local Area Network  
LAN

〃

Wide Area Network  
WAN

〃

Metropolitan Area Network  
MAN

〃

VLAN, CAN, PAN 등등  
기타

〃

〃

〃

〃

# 네트워크의 분류

크기에 따른 분류



Local Area Network  
LAN



Wide Area Network  
WAN



# 네트워크의 분류

## 크기에 따른 분류

“

Local Area Network  
LAN

“

LAN은 가까운 지역을 하나로 묶은 네트워크

# 네트워크의 분류

크기에 따른 분류

“

Wide Area Network  
WAN

”

WAN은 멀리 있는 지역을 한데 묶은 네트워크

가까운 지역끼리 묶인 LAN과 LAN을  
다시 하나로 묶은 것

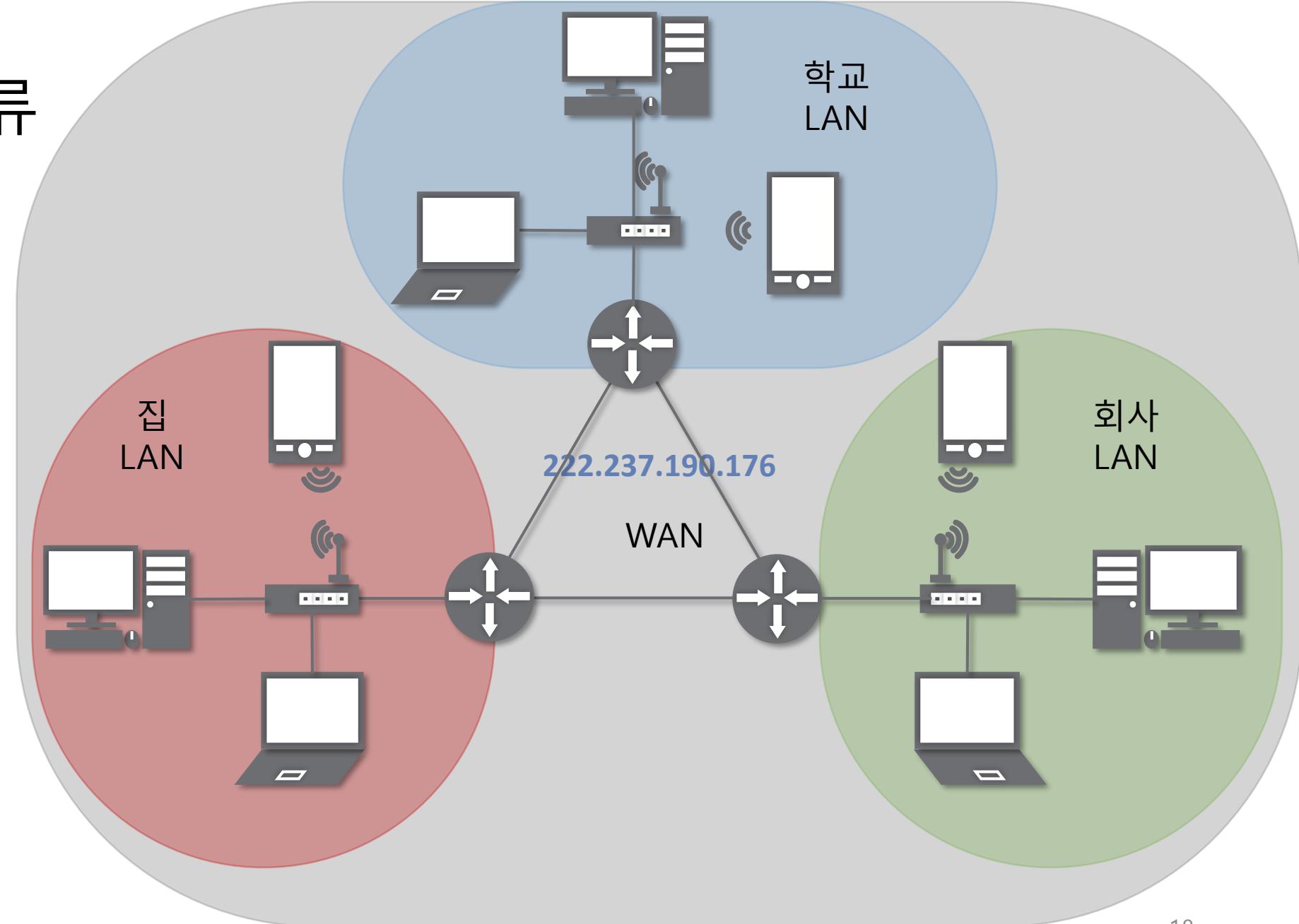
# 네트워크의 분류

크기에 따른 분류

〃

Wide Area Network  
WAN

〃



# 네트워크의 분류

연결 형태에 따른 분류

“

중앙 장비에  
모든 노드가 연결된  
Star 형

“

여려 노드들이  
서로 그물처럼 연결된  
Mesh형

“

마치 나무의 가지처럼  
계층 구조로 연결된  
Tree 형

“

링형, 버스형, 혼합형 등등  
기타

“

“

“

“

# 네트워크의 분류

연결 형태에 따른 분류

---

〃

중앙 장비에  
모든 노드가 연결된  
Star 형

〃

여려 노드들이  
서로 그물처럼 연결된  
Mesh형

〃

---

〃

---

# 네트워크의 분류

연결 형태에 따른 분류

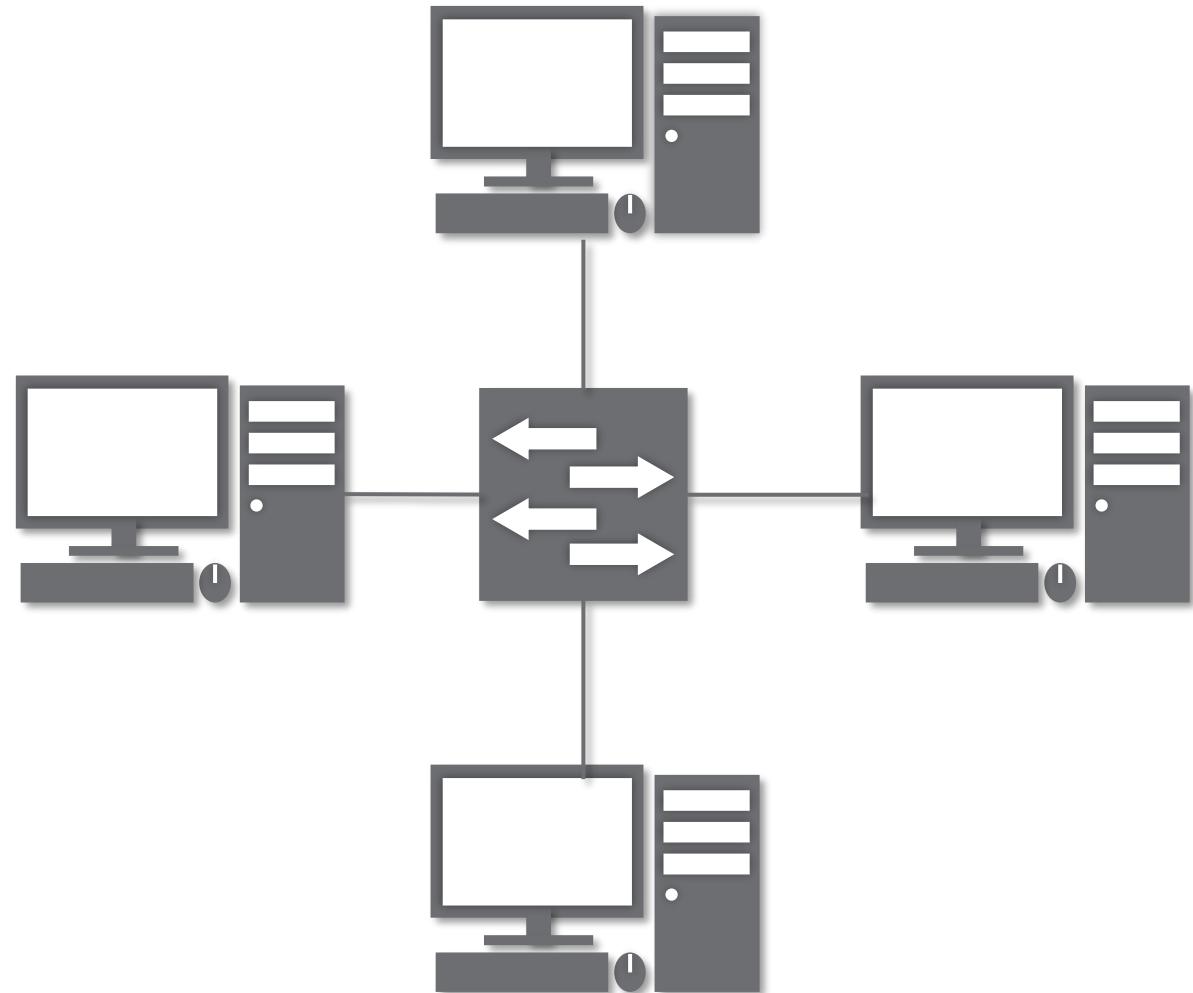
---

"

중앙 장비에  
모든 노드가 연결된  
Star 형

"

---



# 네트워크의 분류

연결 형태에 따른 분류

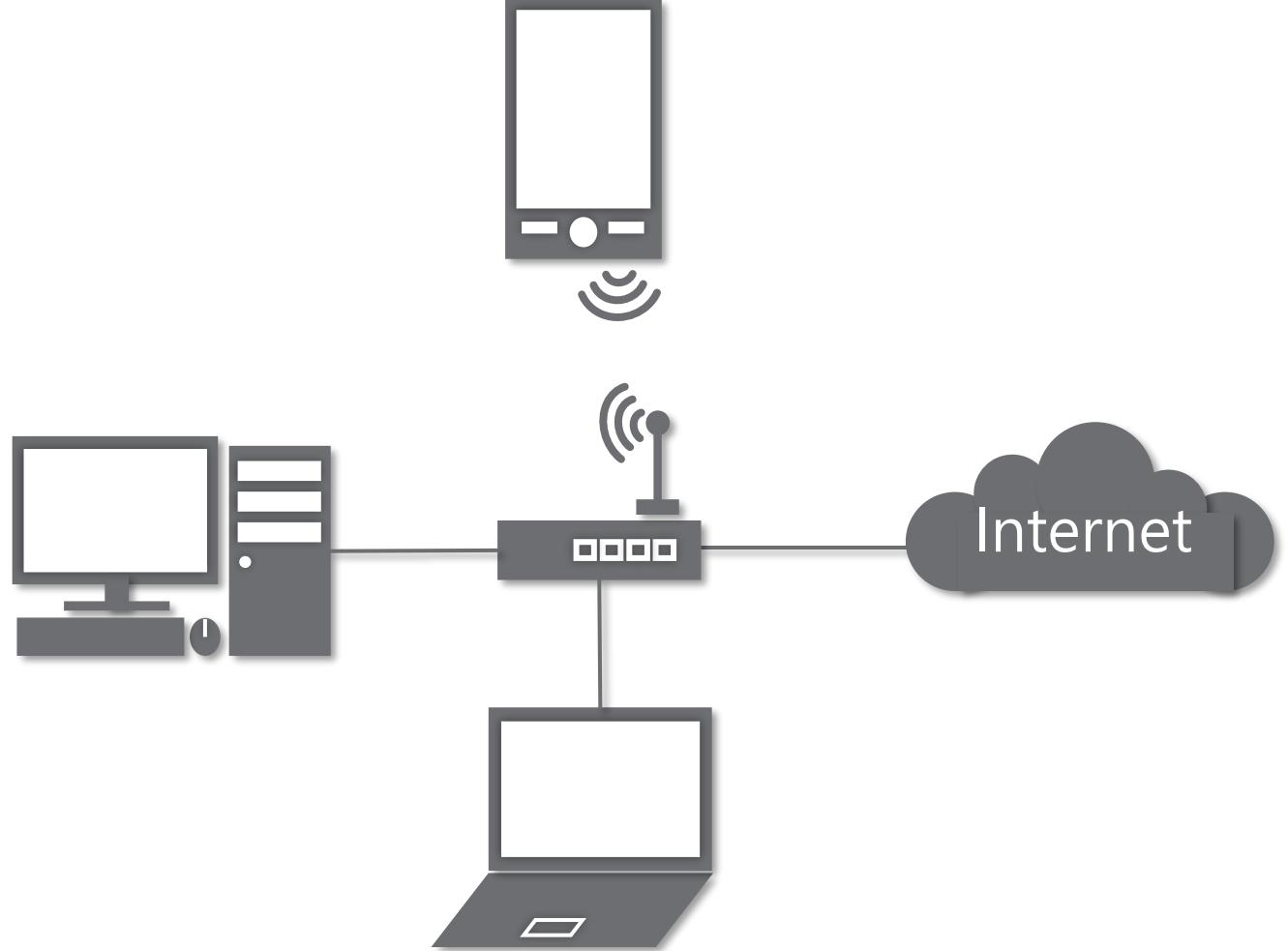
---

“

중앙 장비에  
모든 노드가 연결된  
Star 형

”

---



일반적으로 가정집에서는 공유기를 통해서  
핸드폰, 컴퓨터, TV 등등이 연결된다.  
이 때 만약 공유기가 고장난다면??

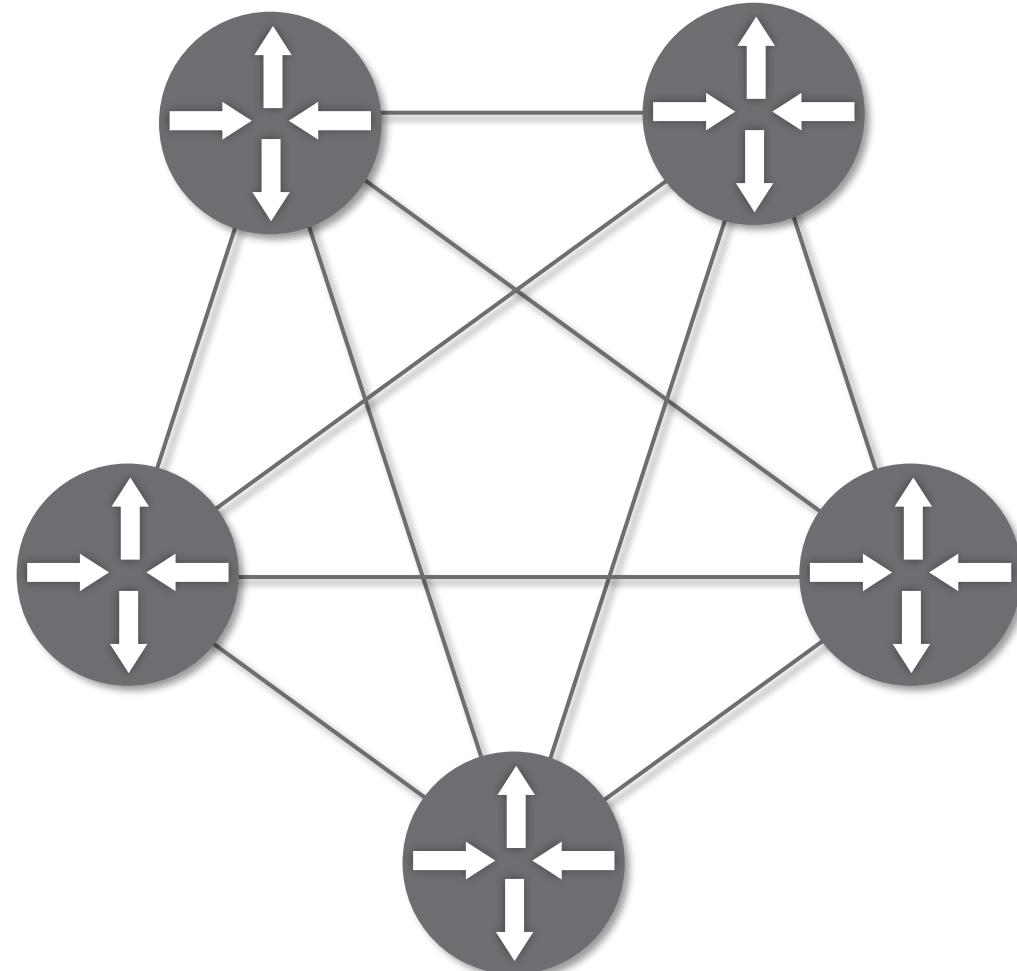
# 네트워크의 분류

연결 형태에 따른 분류

〃

여러 노드들이  
서로 그물처럼 연결된  
Mesh형

〃



# 네트워크의 분류

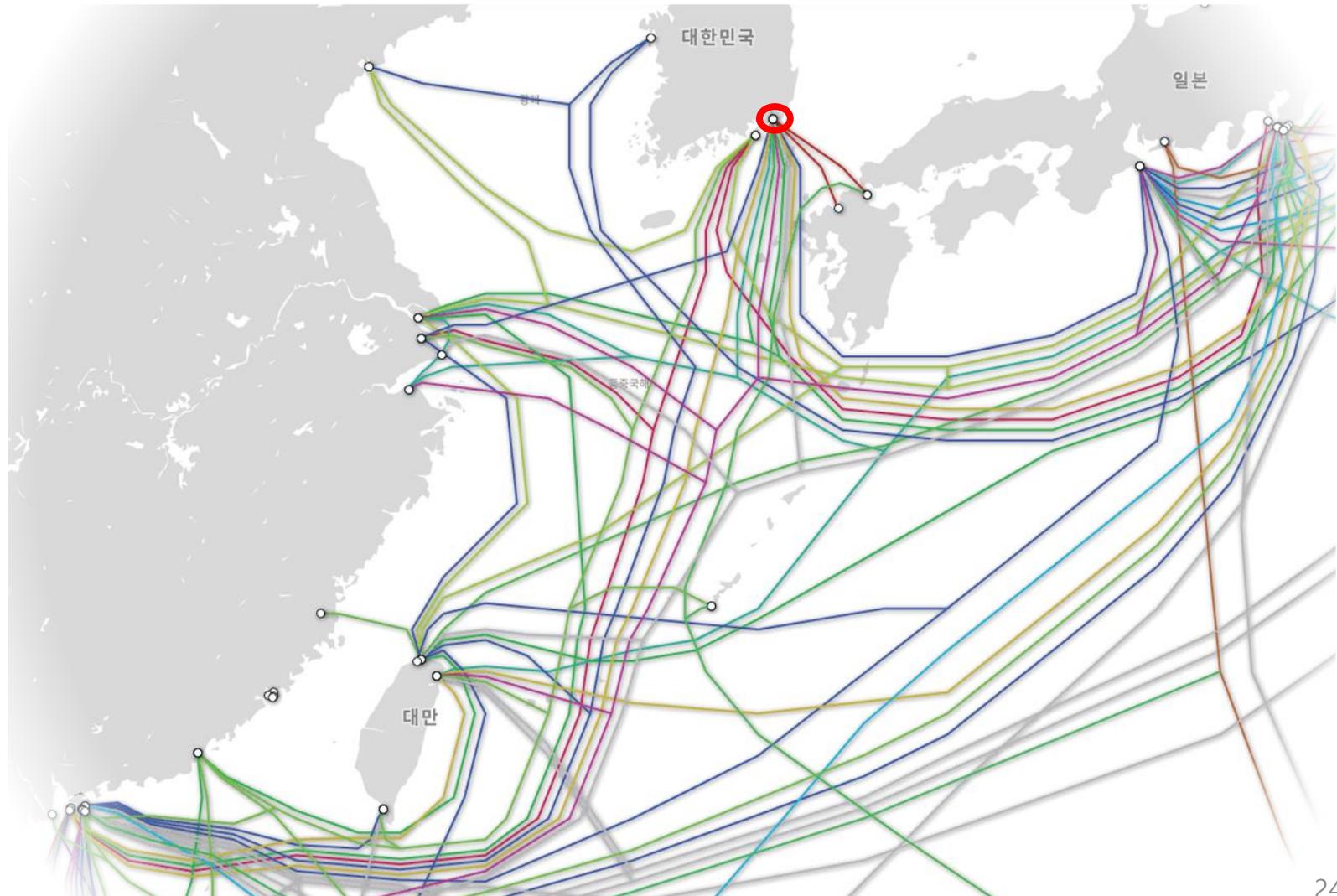
연결 형태에 따른 분류

〃

여러 노드들이  
서로 그물처럼 연결된  
Mesh형

〃

실제 우리나라가 다른 나라와 연결되어 있는 형태  
<https://www.submarinecablemap.com/#/>



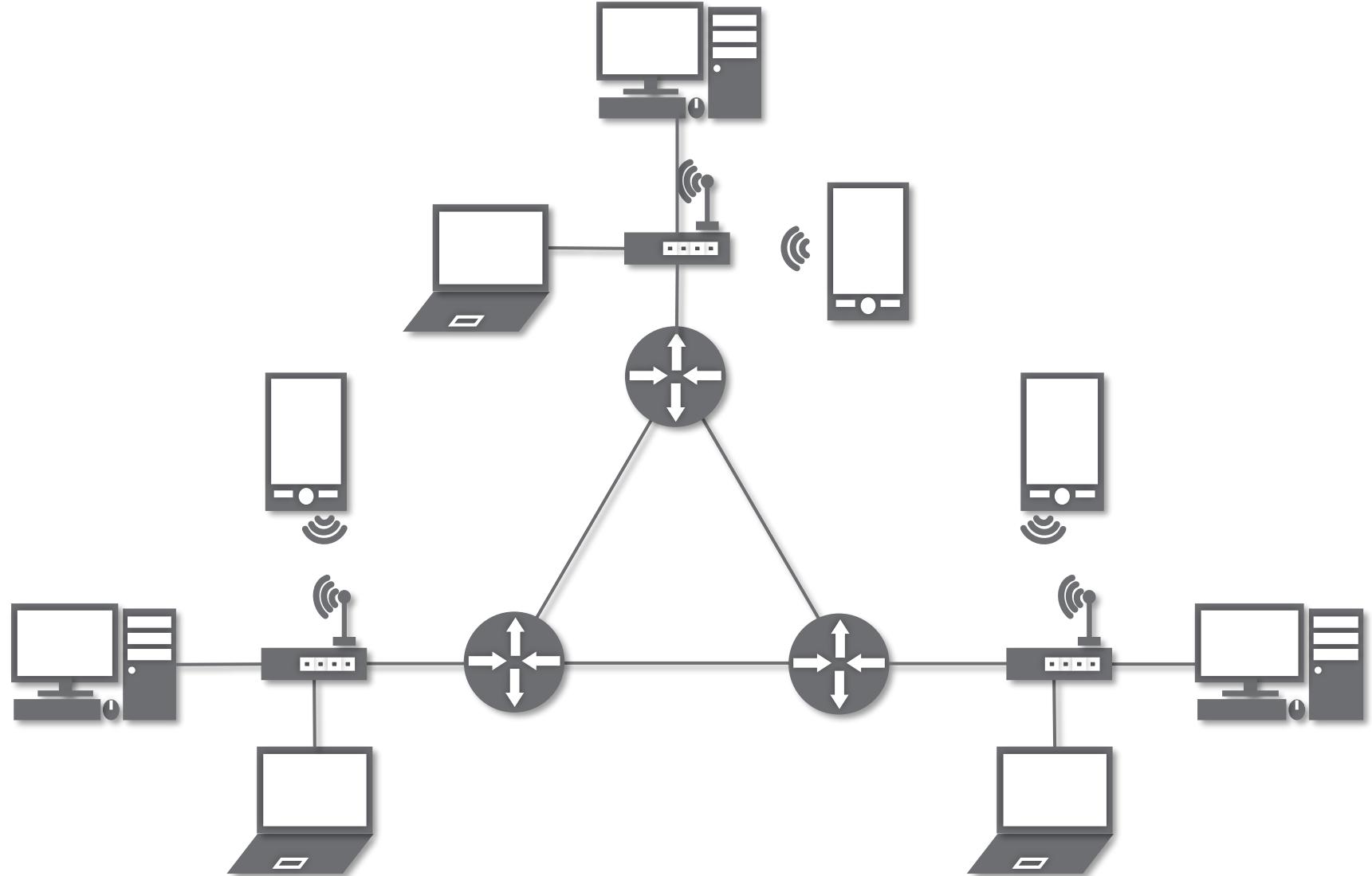
# 네트워크의 분류

연결 형태에 따른 분류

"

실제 인터넷은  
여러 형태를 혼합한 형태  
**혼합형**

"



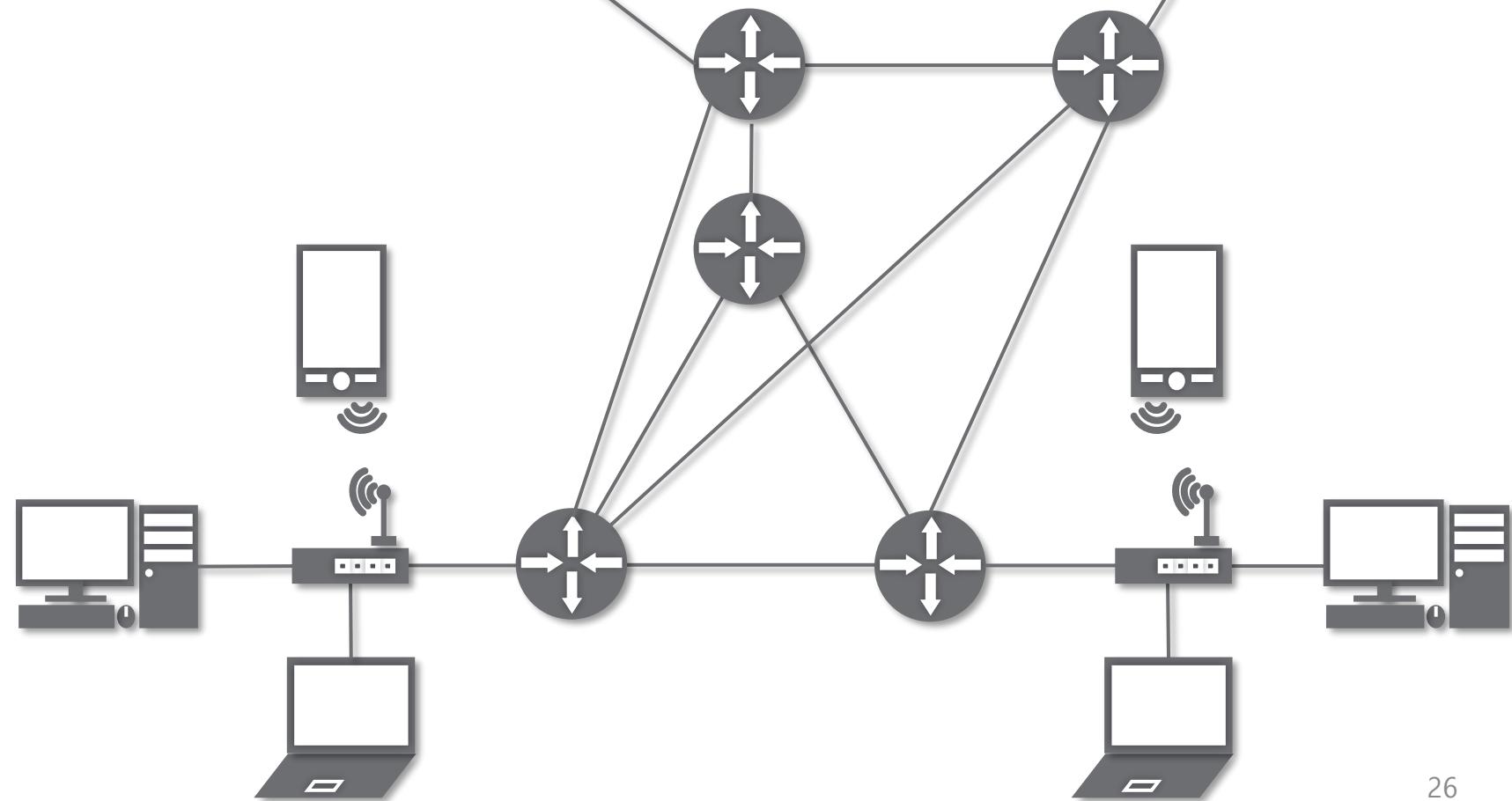
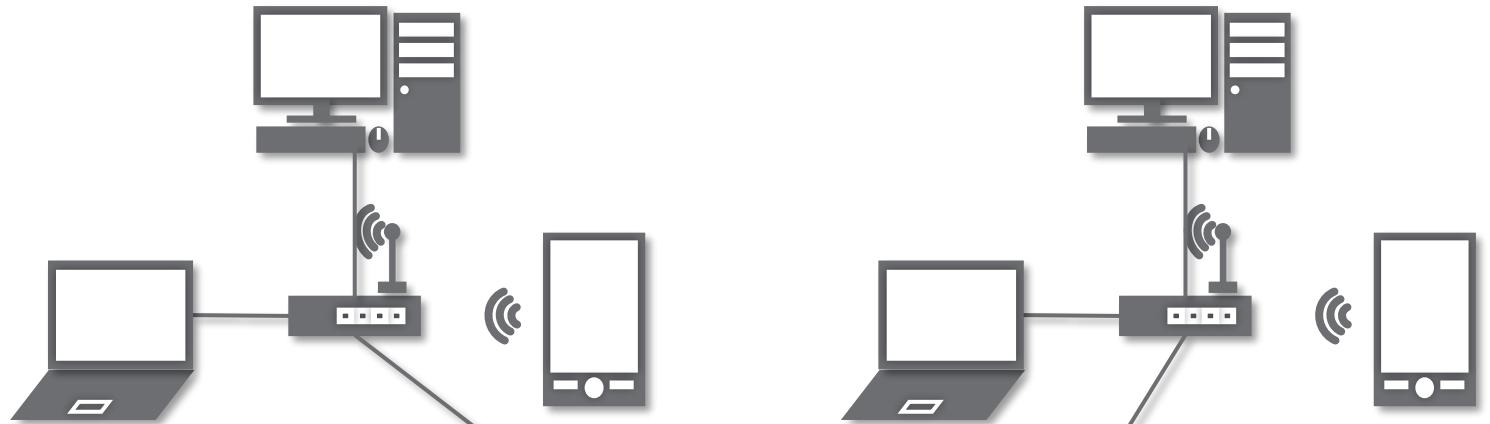
# 네트워크의 분류

연결 형태에 따른 분류

〃

실제 인터넷은  
여러 형태를 혼합한 형태  
**혼합형**

〃





# 네트워크의 통신 방식

# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

---

“

특정 대상이랑만  
1:1로 통신하는  
**유니 캐스트**

“

특정 다수와  
1:N으로 통신하는  
**멀티 캐스트**

“

네트워크에 있는  
모든 대상과 통신하는  
**브로드 캐스트**

“

“

“

# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

“

특정 대상이랑만  
1:1로 통신하는  
유니 캐스트

“



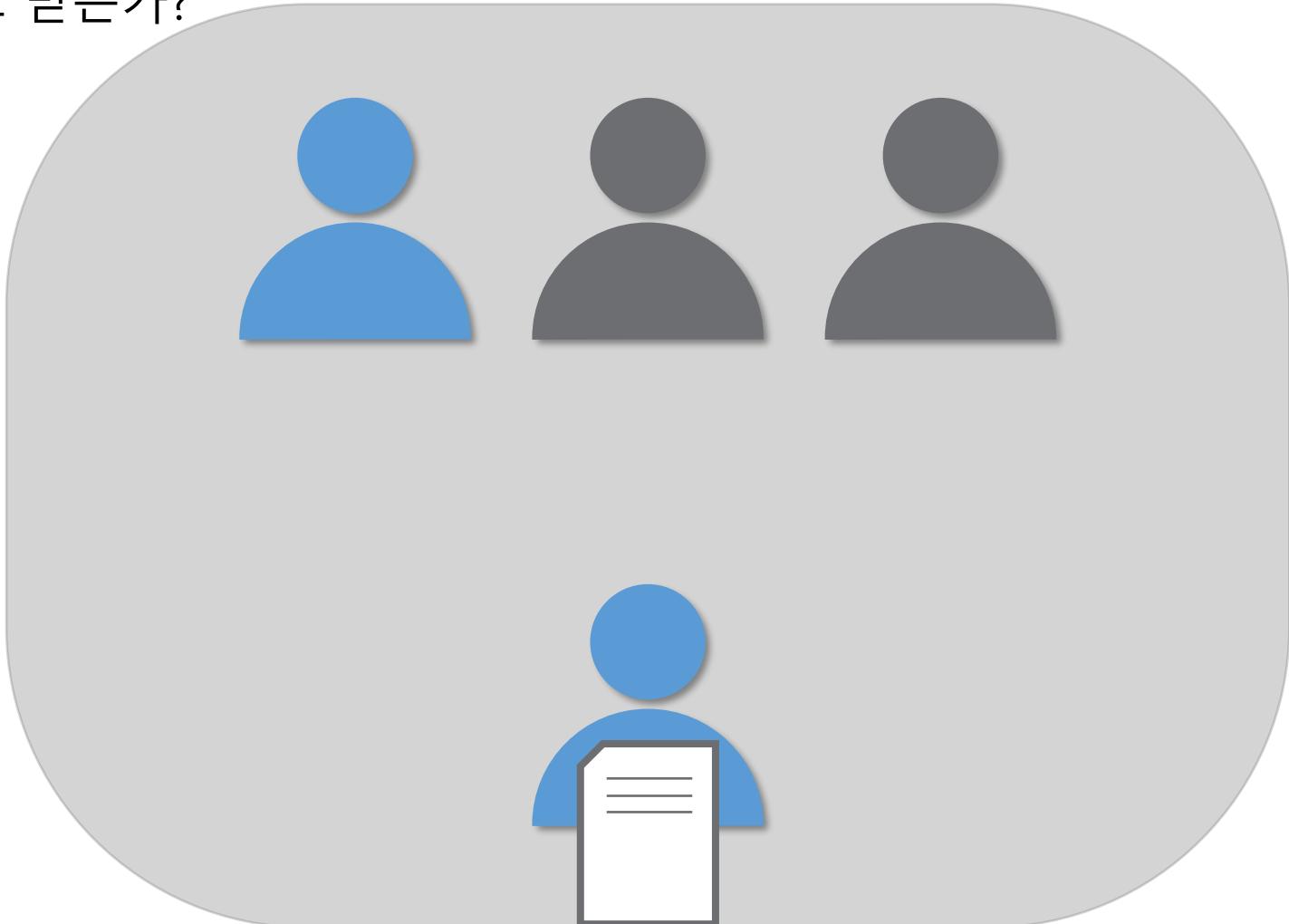
# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

“

특정 대상이랑만  
1:1로 통신하는  
유니 캐스트

“



# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

〃

특정 다수와  
1:N으로 통신하는  
멀티 캐스트

〃



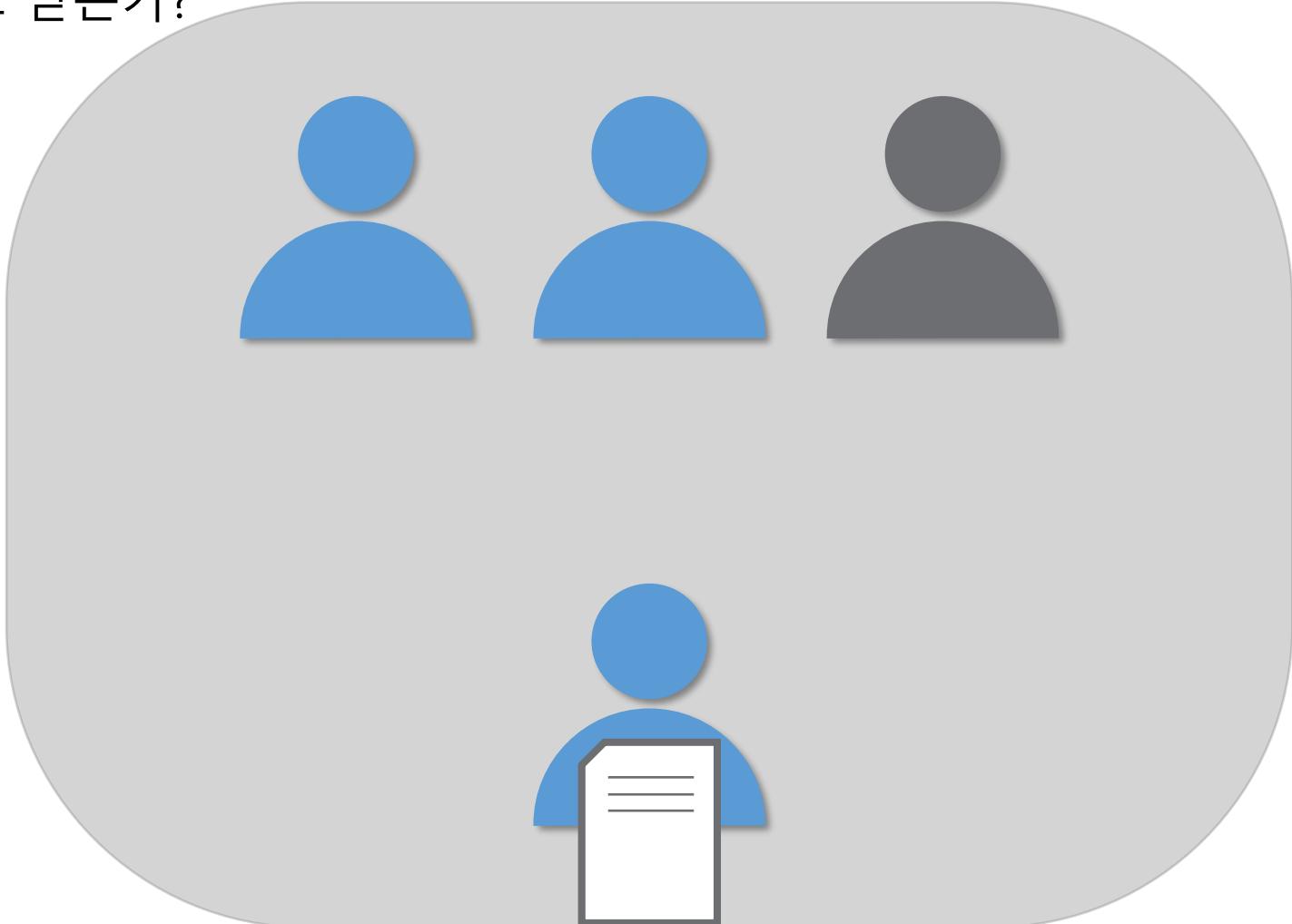
# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

〃

특정 다수와  
1:N으로 통신하는  
멀티 캐스트

〃



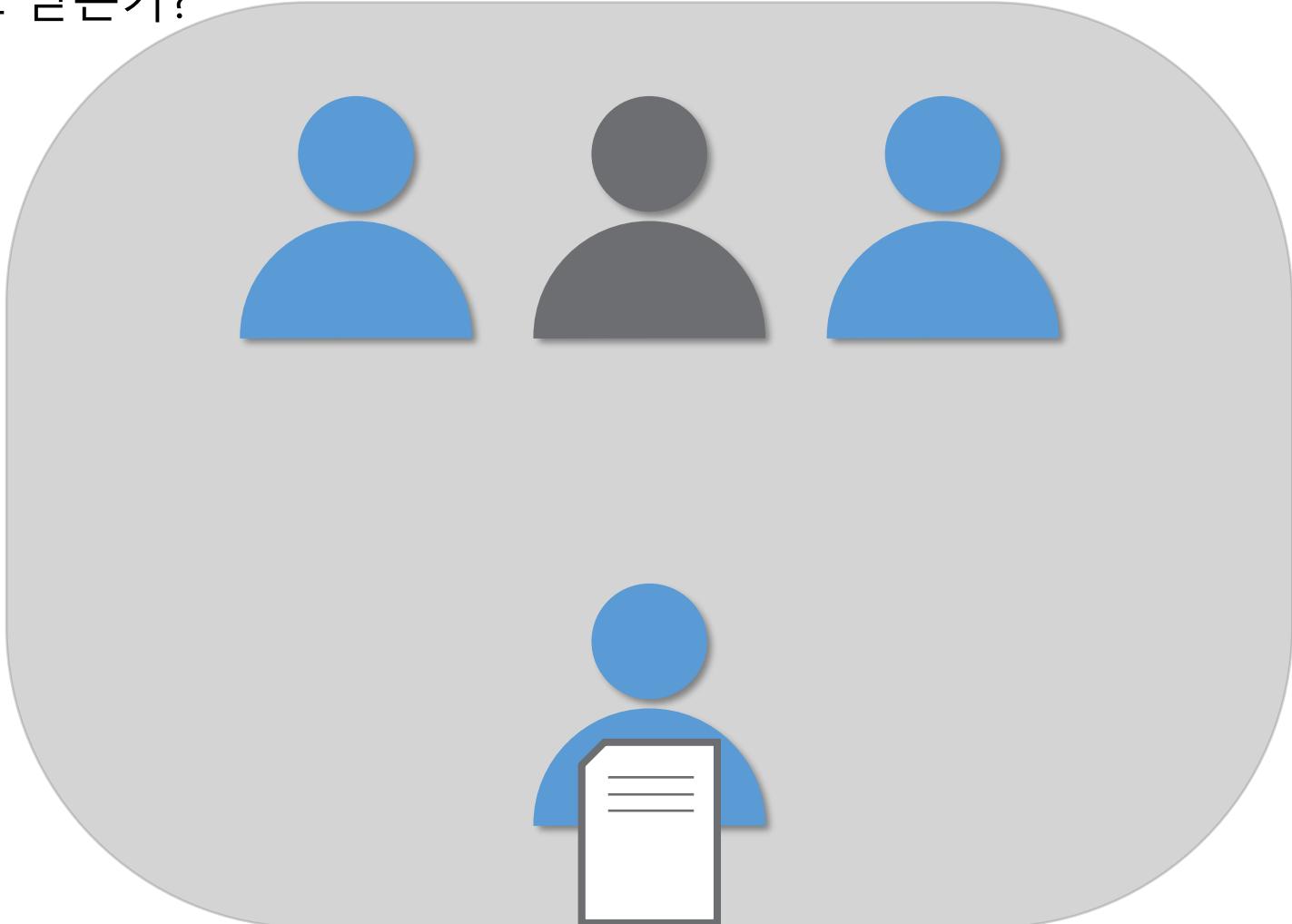
# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

〃

특정 다수와  
1:N으로 통신하는  
멀티 캐스트

〃



# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

“

네트워크에 있는  
모든 대상과 통신하는  
브로드 캐스트

”



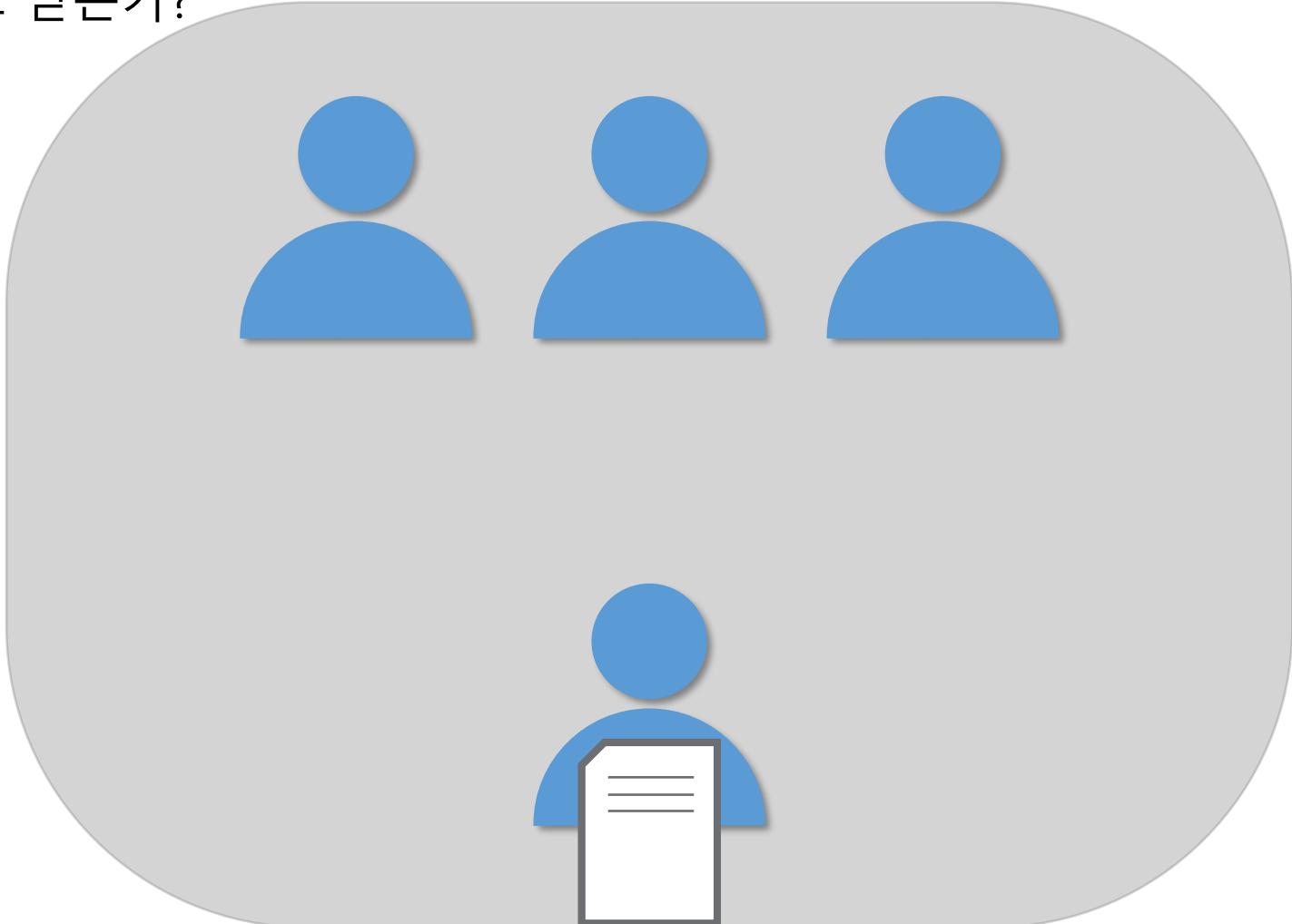
# 네트워크의 통신방식

네트워크에서 데이터는 어떻게 주고 받는가?

“

네트워크에 있는  
모든 대상과 통신하는  
브로드 캐스트

”



# **네트워크 프로토콜**

# 네트워크 프로토콜

네트워크에서 데이터는 어떻게 주고 받는가?

“

네트워크에 있는  
특정한 사용자를 어떻게 찾아낼까?

”



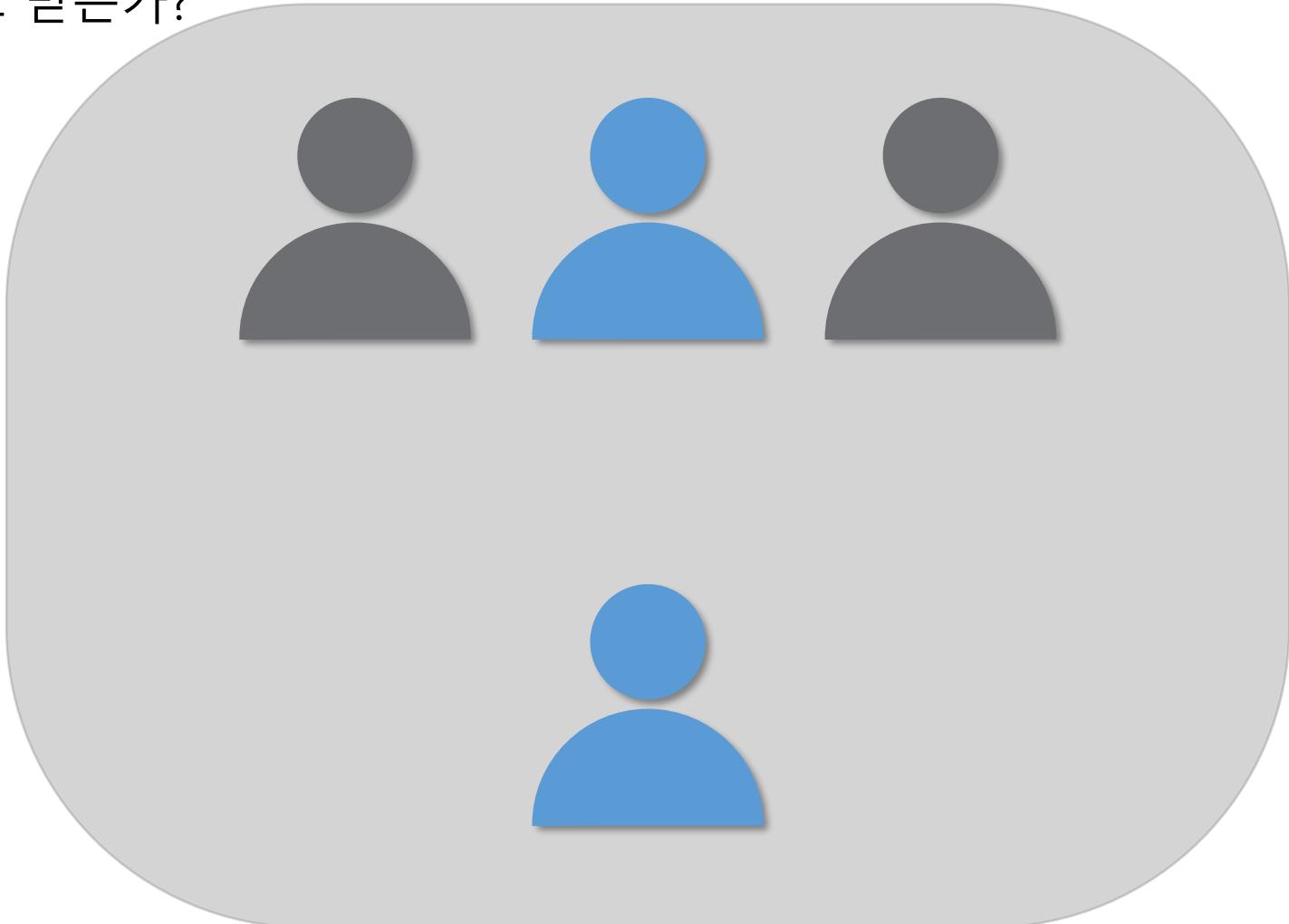
# 네트워크 프로토콜

네트워크에서 데이터는 어떻게 주고 받는가?

“

네트워크에 있는  
특정한 사용자를 어떻게 찾아낼까?

”



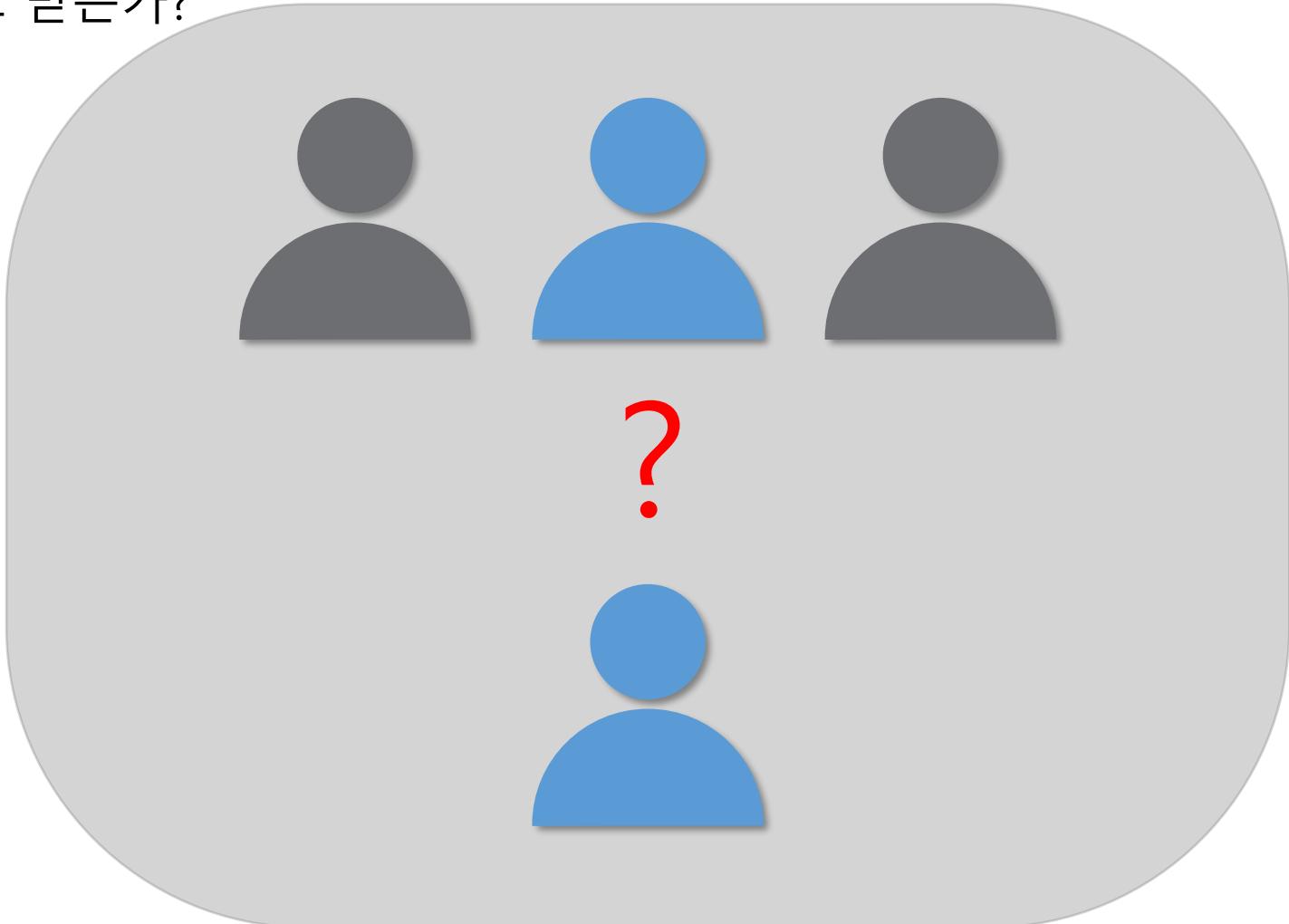
# 네트워크 프로토콜

네트워크에서 데이터는 어떻게 주고 받는가?

“

네트워크에 있는  
특정한 사용자를 어떻게 찾아낼까?

”



# 네트워크 프로토콜 프로토콜이란?

프로토콜은 일종의 약속, 양식

네트워크에서  
노드와 노드가 통신할 때  
어떤 노드가 어느 노드에게  
어떤 데이터를 어떻게 보내는지  
작성하기 위한 양식

택배는 택배만의 양식  
편지는 편지만의 양식  
전화는 전화만의 양식

각 프로토콜들도 해당  
프로토콜만의 양식



# 네트워크 프로토콜

## 프로토콜이란?

프로토콜은 일종의 약속, 양식

네트워크에서  
노드와 노드가 통신할 때  
어떤 노드가 어느 노드에게  
어떤 데이터를 어떻게 보내는지  
작성하기 위한 양식

택배는 택배만의 양식  
편지는 편지만의 양식  
전화는 전화만의 양식

각 프로토콜들도 해당  
프로토콜만의 양식



# 네트워크 프로토콜

## 여러가지 프로토콜

가까운 곳과 연락할 때

(Ethernet 프로토콜  
(MAC 주소))

멀리 있는 곳과 연락할 때

(ICMP  
IPv4  
ARP  
(IP 주소))

여러가지 프로그램으로 연락할 때

(TCP, UDP  
(포트 번호))

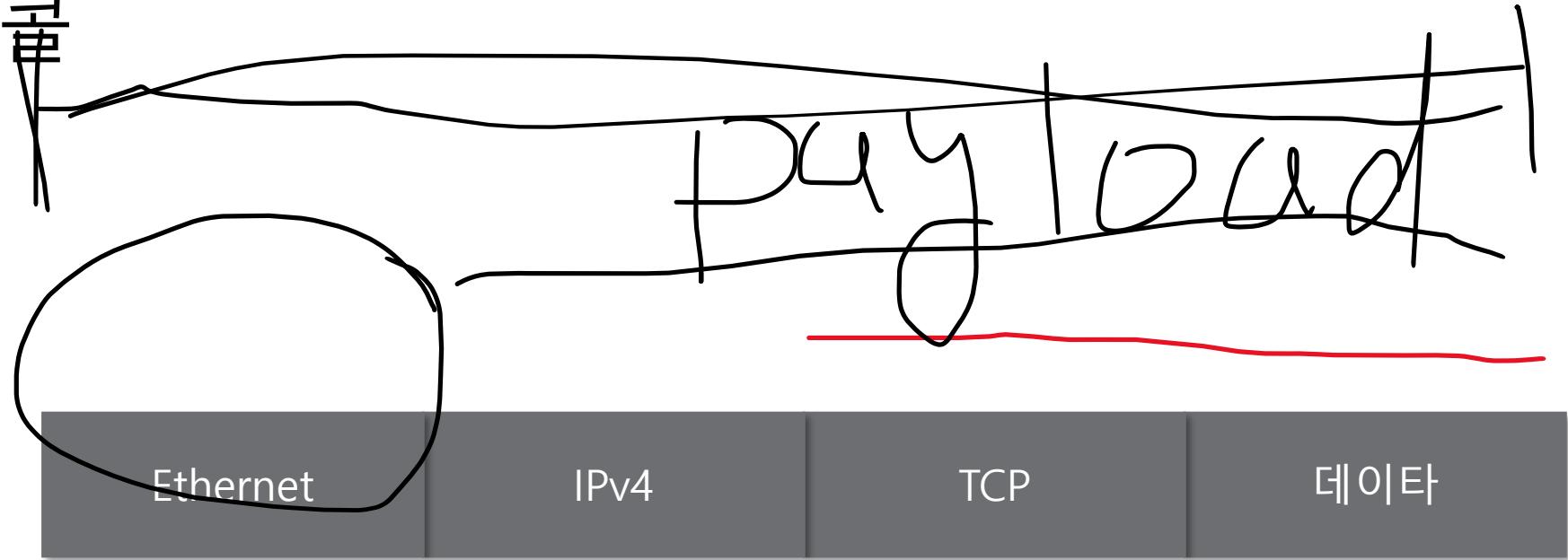
# 네트워크 프로토콜

여러가지 프로토콜

" "

여러 프로토콜들로  
캡슐화 된  
패킷

" "



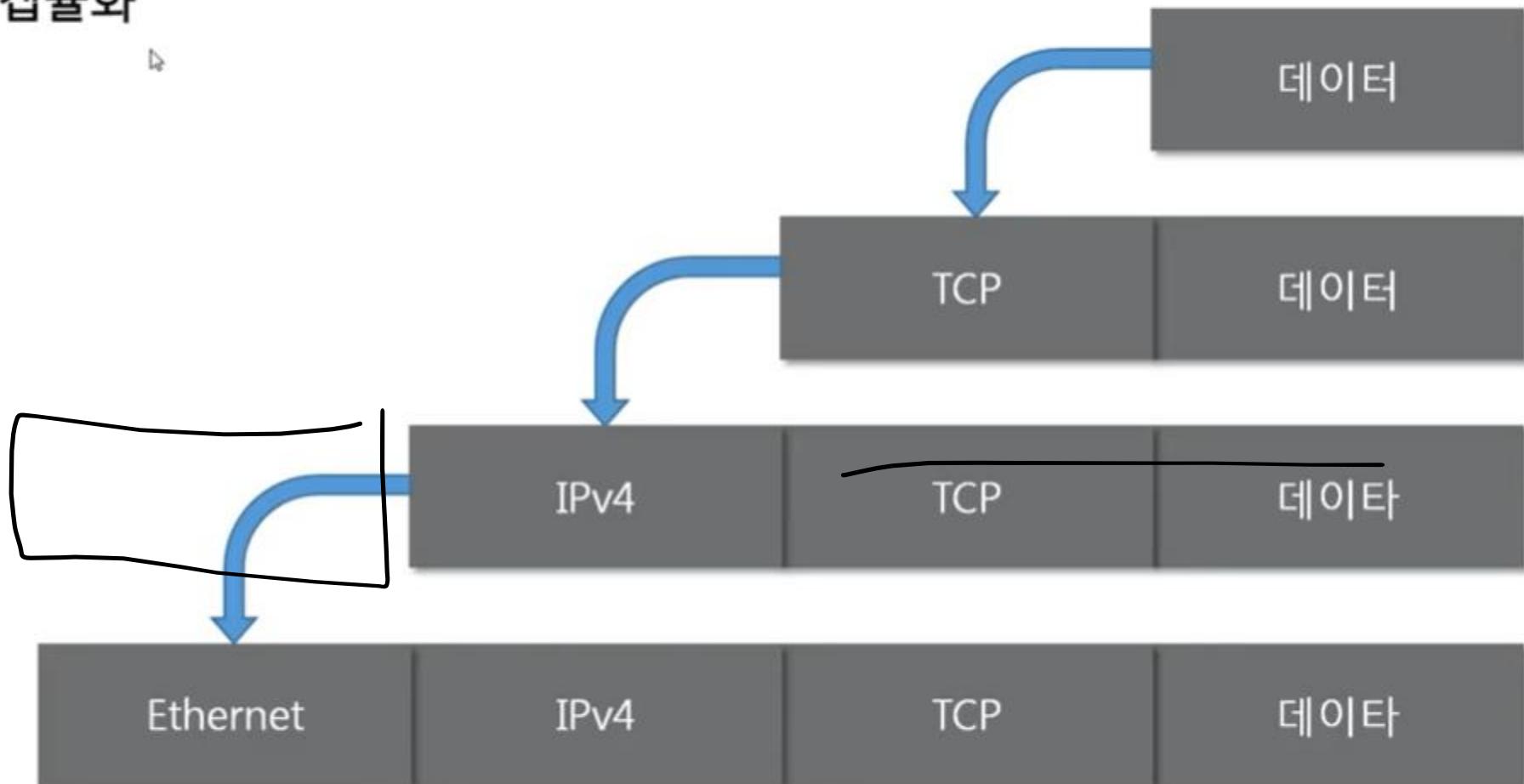
# 네트워크를 통해 전달되는 데이터, 패킷

패킷을 이용한 통신과정 - 캡슐화

“

여러 프로토콜을 이용해서  
최종적으로 보낼 때  
패킷을 만드는 과정

”

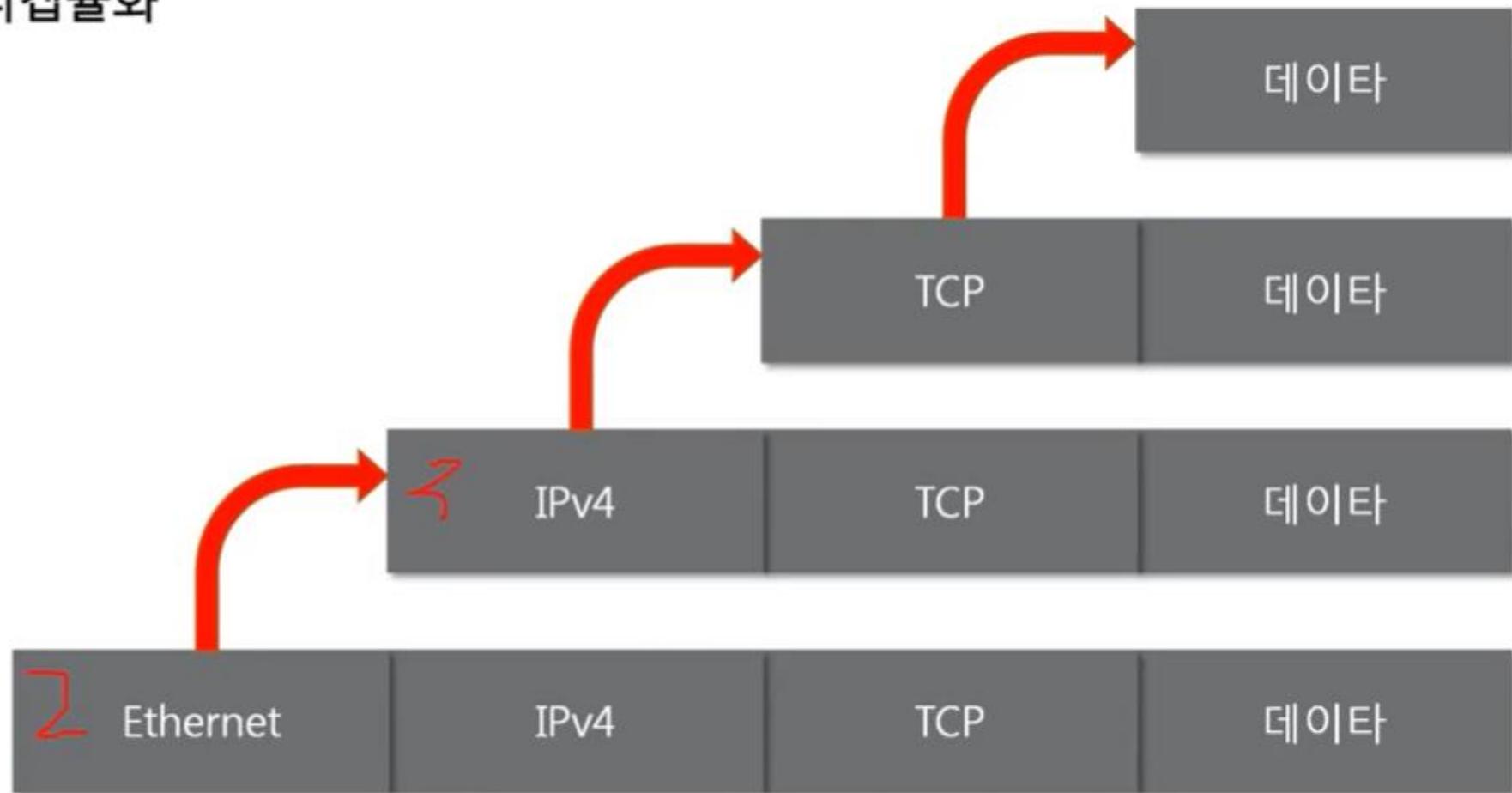


# 네트워크를 통해 전달되는 데이터, 패킷

패킷을 이용한 통신과정 - 디캡슐화

“

패킷을 받았을 때  
프로토콜들을 하나씩 확인하면서  
데이터를 확인하는 과정



”

# 네트워크를 통해 전달되는 데이터, 패킷

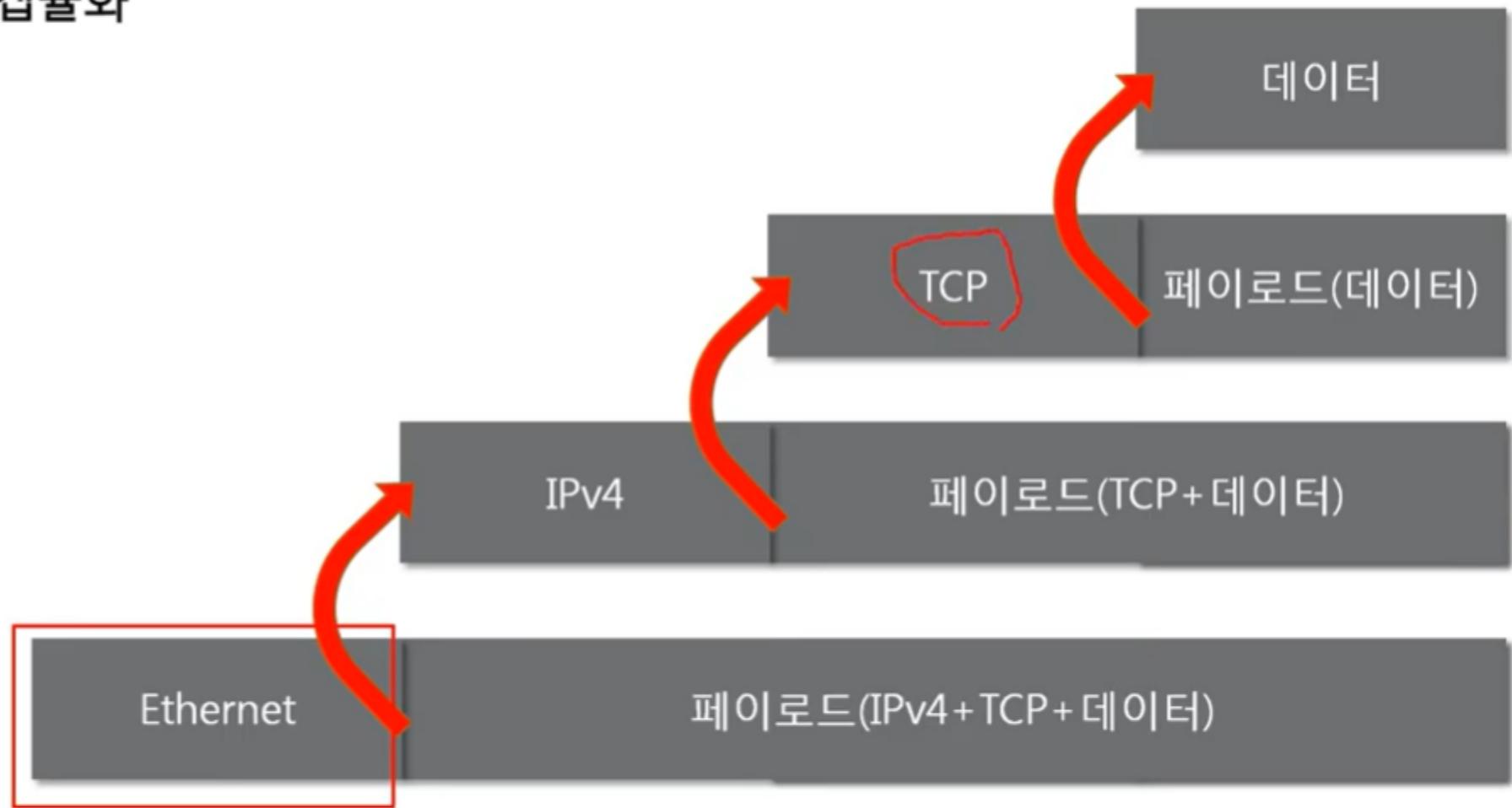
패킷을 이용한 통신과정 - 캡슐화

“

여러 프로토콜을 이용해서  
최종적으로 보낼 때

패킷을 만드는 과정

”

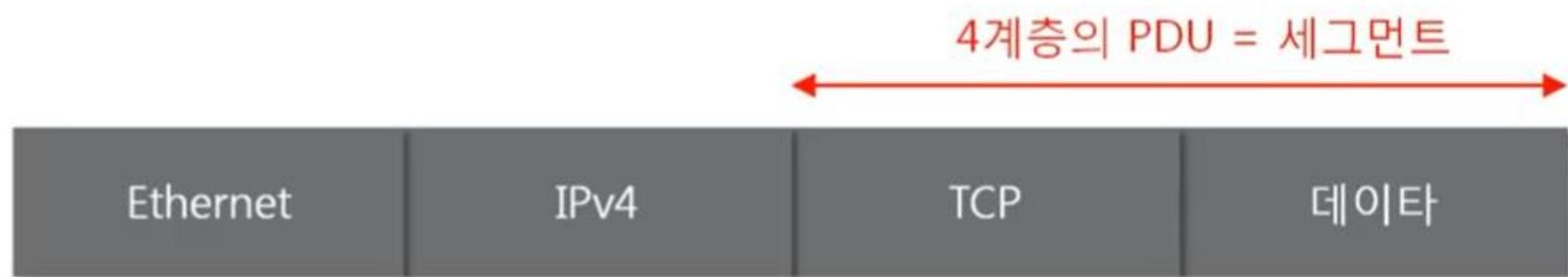


# 네트워크를 통해 전달되는 데이터, 패킷

계층별 패킷의 이름 PDU

〃

계층별로 이름이 다른  
PDU  
Protocol Data Unit



〃

# 네트워크를 통해 전달되는 데이터, 패킷

계층별 패킷의 이름 PDU

〃

계층별로 이름이 다른  
PDU

3계층의 PDU = 패킷



〃

# **네트워크 프로토콜 실습**

# 실습과제

## 1. 구글과 나는 어떻게 연결되어 있는지 확인해보기

구글의 서버와 여러분의 컴퓨터가 어떻게 연결되어 있는지 확인해보기

## 2. Wireshark 설치

프로토콜이 어떻게 생겼는지 직접 보기 위해 사용할 프로그램을 설치

## 3. 프로토콜 직접 확인해보기

설치한 Wireshark를 이용해서 현재 컴퓨터에서 인터넷을 통해 주고받고 있는 모든 내용을 직접 확인해보기

```
명령 프롬프트

C:\Users\oaky\>tracert 8.8.8.8

최대 30홉 이상의
dns.google [8.8.8.8] (으)로 가는 경로 추적:

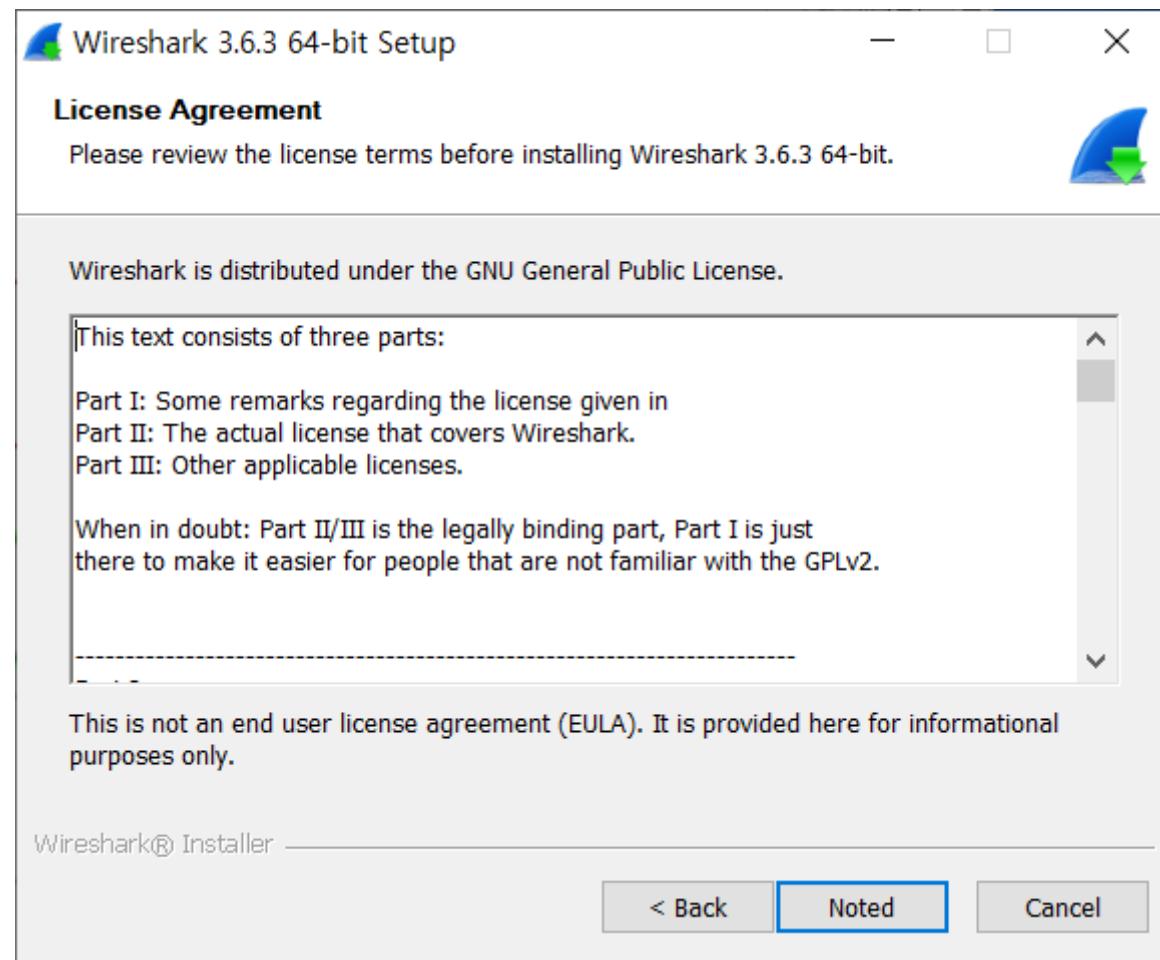
 1   9 ms    3 ms    3 ms  192.168.0.1
 2   *         *         *   요청 시간이 만료되었습니다.
 3   3 ms    4 ms    2 ms  100.71.26.177
 4   4 ms    3 ms    4 ms  10.44.254.46
 5   4 ms    3 ms    3 ms  10.222.18.140
 6   8 ms    3 ms    3 ms  10.222.23.207
 7  39 ms   41 ms   37 ms  72.14.196.26
 8  36 ms   36 ms   37 ms  142.251.52.31
 9  39 ms   38 ms   38 ms  142.250.226.7
10  37 ms   37 ms   37 ms  dns.google [8.8.8.8]

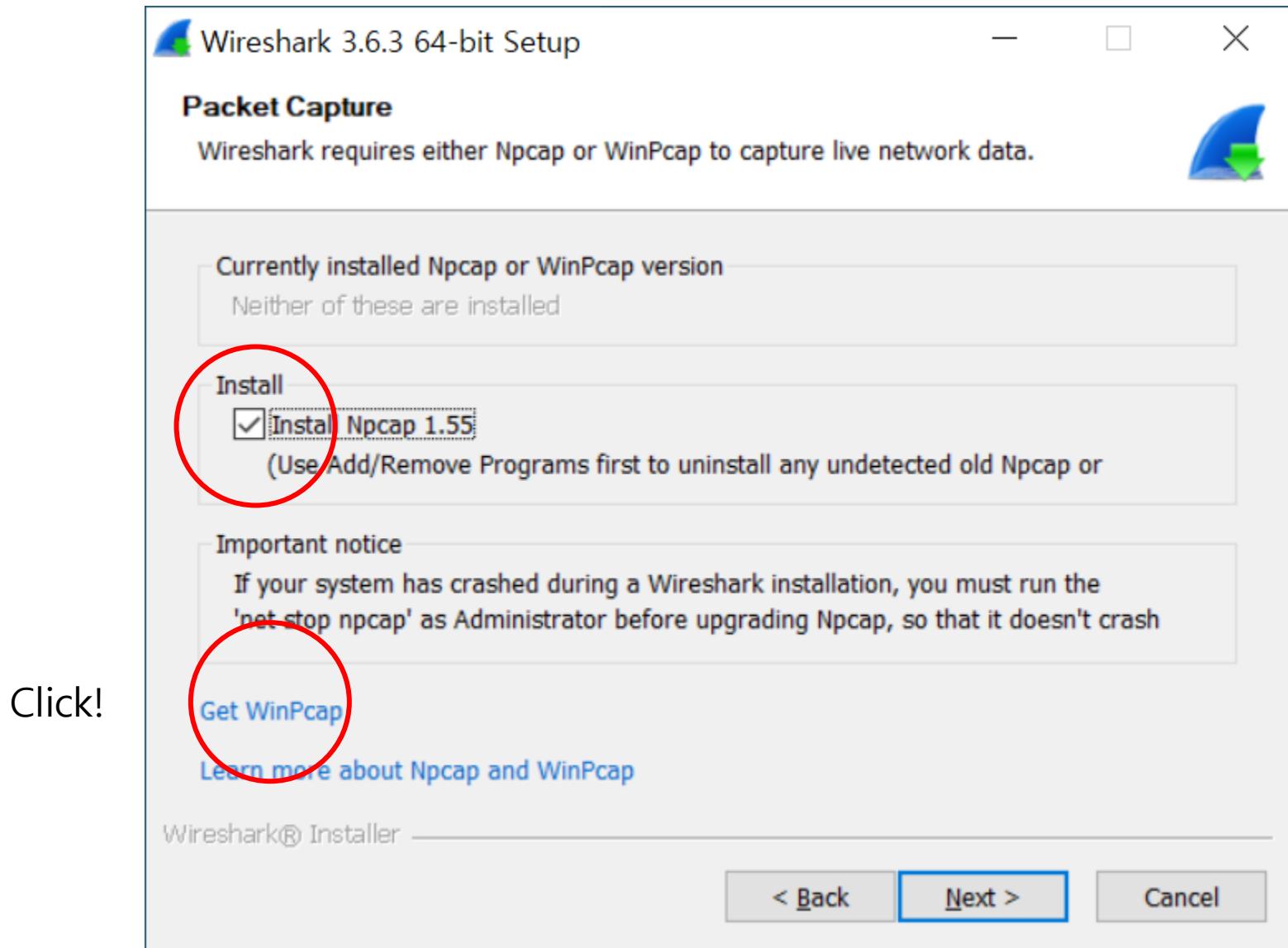
추적을 완료했습니다.

C:\Users\oaky\>
```

# WireShark 설치

<https://www.wireshark.org/download.html>





Click!

Riverbed Technology Wireshark

# WinPcap

*The industry-standard windows packet capture library*

WinPcap | WinDump | NTAR

Search

 Download  
Get WinPcap

 Documentation

 Support

## News and Releases

**15 September 2018**

WinPcap, though still available for download (v4.1.3), has not seen an upgrade in many years and there are no road map/future plans to update the technology. While community support may persist, technical oversight by Riverbed staff, responses to questions posed by Riverbed resources, and bug reporting are no longer available.

Gordon Lyon, Nmap project founder, has created Npcap, a packet capture library for Windows, that includes WinPcap compatibility and may be a suitable replacement for WinPcap and WinPcap Pro. Information can be found at <https://nmap.org/npcap>.

[More...](#)

## Introduction to WinPcap

For many years, WinPcap has been recognized as the industry-standard tool for link-layer network access in Windows environments, allowing applications to capture and transmit network packets bypassing the protocol stack, and including kernel-level packet filtering, a network statistics engine and support for remote [packet capture](#).

WinPcap consists of a driver that extends the operating system to provide low-level network access and a library that is used to easily access low-level network layers. This library also contains the Windows version of the well-known libpcap Unix API.

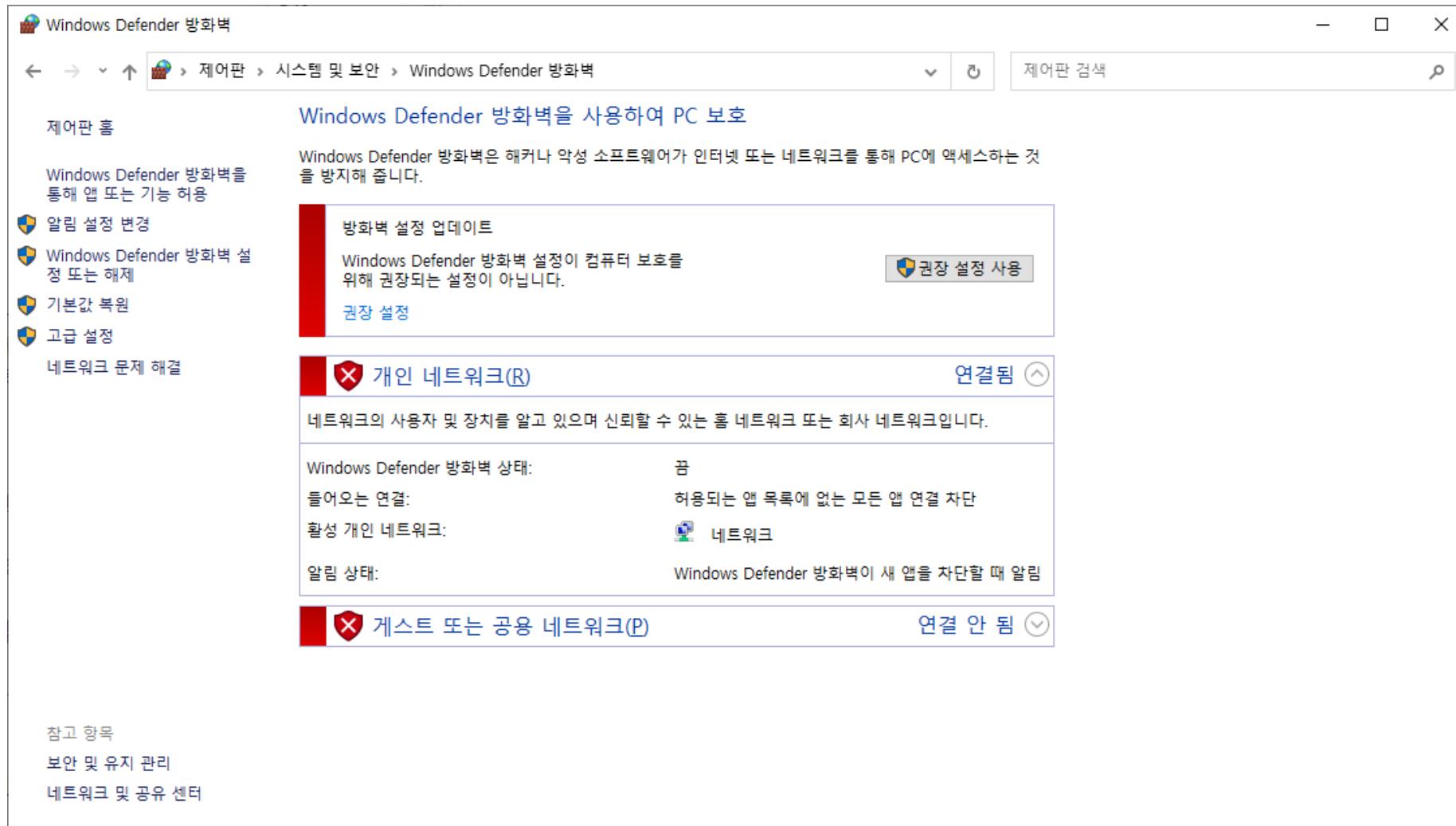
Thanks to its set of features, WinPcap has been the [packet capture](#) and filtering engine for many open source and commercial network tools, including protocol analyzers, network monitors, network intrusion detection systems, sniffers, traffic generators and network testers. Some of these [networking tools](#), like [Wireshark](#), Nmap, Snort, and ntop are known and used throughout the networking community.

Winpcap.org is also the home of [WinDump](#), the Windows version of the popular tcpdump tool. WinDump can be used to watch, diagnose and save to disk network traffic according to various complex rules.

Copyright © 2018 Riverbed Technology • [Privacy Policy](#) | [Legal Notices](#)

The screenshot shows the official WinPcap website. At the top, there's a dark green header bar with the Riverbed Technology logo and a Wireshark link. Below the header, the main title "WinPcap" is prominently displayed in a large, white, sans-serif font. Underneath the title is a navigation bar with three tabs: "WinPcap" (which is active and highlighted in white), "WinDump", and "NTAR". To the right of the tabs is a search bar with a "Search" button. The main content area has a light blue background. It features a bold warning message: "WinPcap Has Ceased Development. We recommend Npcap." Below this, a smaller text states: "The WinPcap project has ceased development and WinPcap and WinDump are no longer maintained. WE RECOMMEND USING Npcap INSTEAD." A note follows: "If you do insist upon using WinPcap, be aware that its installer". A bulleted list provides reasons: "• Uses [NDIS 5.0](#), which might not work well with newer versions of Windows.  
• Was built with an [old version of NSIS](#) and as a result is vulnerable to [DLL hijacking](#)." Further down, it says: "The last official WinPcap release was 4.1.3" and provides a link to the "changelog". It also links to the "Version 4.1.3 Installer for Windows" and "Driver +DLLs". A section titled "Supported platforms:" lists: "• NONE. WinPcap is completely unsupported, and might have compatibility issues with current versions of Windows." Another section titled "Previously supported platforms:" lists: "• Windows NT4/2000  
• Windows XP/2003/Vista/2008/Win7/2008R2/Win8 (x86 and x64)". At the bottom of the page, there are MD5 and SHA1 checksums: "MD5 Checksum: a11a2f0cfe6d0b4c50945989db6360cd" and "SHA1 Checksum: e2516fd1573e70334c8f50bee5241cdfdf48a00". The footer contains copyright information: "Copyright © 2018 Riverbed Technology • [Privacy Policy](#) | [Legal Notices](#)" and a note: "Last modified: Sunday, March 28, 2021".

1. 시작 → 제어판 → 시스템 및 보안 → Windows 방화벽 클릭하세요. 2. 좌측 메뉴에서 Windows 방화벽 설정 또는 해제를 클릭하세요.





## 각 네트워크 유형 설정의 사용자 지정

사용하는 각 네트워크 종류의 방화벽 설정을 수정할 수 있습니다.

### 개인 네트워크 설정



#### ○ Windows Defender 방화벽 사용

- 허용되는 앱 목록에 있는 연결을 포함하여 모든 들어오는 연결 차단
- Windows Defender 방화벽이 새 앱을 차단할 때 알림



#### ● Windows Defender 방화벽 사용 안 함(권장하지 않음)

### 공용 네트워크 설정



#### ○ Windows Defender 방화벽 사용

- 허용되는 앱 목록에 있는 연결을 포함하여 모든 들어오는 연결 차단
- Windows Defender 방화벽이 새 앱을 차단할 때 알림



#### ● Windows Defender 방화벽 사용 안 함(권장하지 않음)

내 PC의 클라이언트 IP 주소는?

<https://ko.infobyip.com/>

# 네트워크의 기준!

## 네트워크 모델

# 목차

## INDEX



# **네트워크 모델의 종류**

# 네트워크 계층 모델

## TCP/IP 모델

1960년대 말 미국방성의 연구에서 시작되어

1980년대 초 프로토콜 모델로 공개

현재의 인터넷에서 컴퓨터들이

서로 정보를 주고받는데 쓰이는

통신 규약(프로토콜)의 모음이다.

4계층 응용

3계층 전송

2계층 네트워크

1계층 네트워크  
인터페이스

# 네트워크 계층 모델

OSI 7계층

1984년 네트워크 통신을 체계적으로 다루는

ISO에서 표준으로 지정한 모델

데이터를 주고받을 때 데이터 자체의 흐름을

각 구간별로 나눠 놓은 것

7계층 응용

6계층 표현

5계층 세션

4계층 전송

3계층 네트워크

2계층 데이터 링크

1계층 물리

# OSI 7계층 모델

OSI 7계층 모델의 계층별 프로토콜

7계층 응용	HTTP, SMTP, IMAP, POP, SNMP, FTP, TELNET, SSH
6계층 표현	SMB, AFP, XDR
5계층 세션	NetBIOS
4계층 전송	TCP, UDP, SPX
3계층 네트워크	IP, ICMP, IGMP, X.25, CLNP, ARP, RARP, BGP, OSPF, RIP, IPX, DDP
2계층 데이터 링크	이더넷, 토큰링, PPP, HDLC, 프레임 릴레이, ISDN, ATM, 무선랜, FDDI
1계층 물리	전선, 전파, 광섬유, 동축케이블, 도파관, PSTN, 리피터, DSU, CSU, 모뎀

# 두 모델 비교

# 두 모델 비교

## 공통점과 차이점

4계층 응용

3계층 전송

2계층 네트워크

1계층 네트워크  
인터페이스

### 공통점

계층적 네트워크 모델  
계층간 역할 정의

### 차이점

계층의 수 차이  
OSI는 역할 기반, TCP/IP는 프로토콜 기반  
OSI는 통신 전반에 대한 표준  
TCP/IP는 데이터 전송기술 특화

7계층 응용

6계층 표현

5계층 세션

4계층 전송

3계층 네트워크

2계층 데이터 링크

1계층 물리

# TCP/IP 모델

## TCP/IP 모델의 특징

1960년대 말 미국방성의 연구에서 시작되어

1980년대 초 프로토콜 모델로 공개

현재의 인터넷에서 컴퓨터들이

서로 정보를 주고받는데 쓰이는

통신 규약(프로토콜)의 모음이다.

4계층 응용

3계층 전송

2계층 네트워크

1계층 네트워크  
인터페이스

**네트워크를 통해  
전달되는 데이터, 패킷**

# 네트워크를 통해 전달되는 데이터, 패킷 패킷이란?

패킷이란 네트워크 상에서 전달되는 데이터를  
통칭하는 말로 네트워크에서 전달하는  
데이터의 형식화된 블록이다.

패킷은 제어 정보와 사용자 데이터로 이루어지며  
사용자 데이터는 페이로드라고도 한다.

4계층 표현

3계층 전송

2계층 네트워크

1계층 데이터 링크

# 네트워크를 통해 전달되는 데이터, 패킷 패킷이란?

“

인형 속의 인형, 마트료시카  
러시아 인형

”

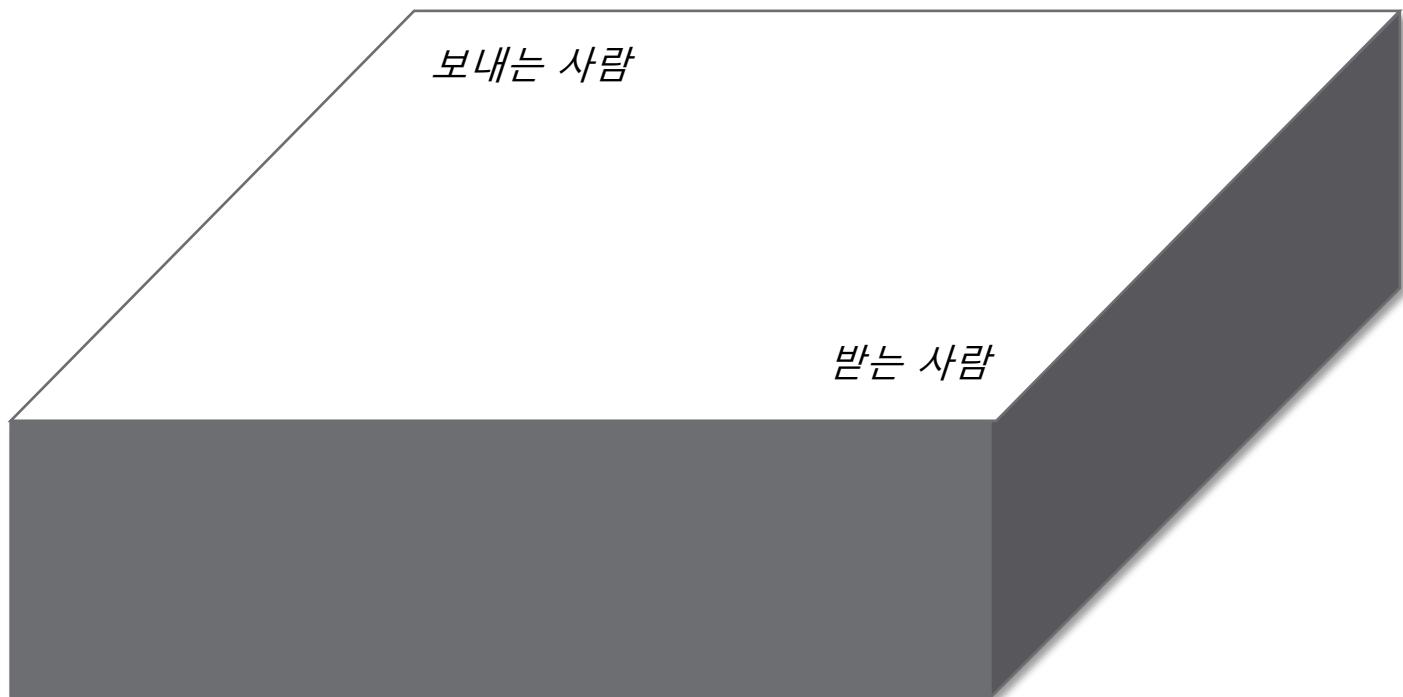


# 네트워크를 통해 전달되는 데이터, 패킷 패킷이란?

여러 번 포장된  
택배 상자

“

“

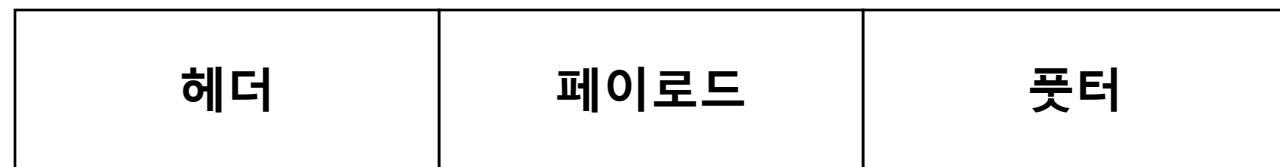


# 네트워크를 통해 전달되는 데이터, 패킷 패킷이란?

---

“

여러 프로토콜들로  
캡슐화 된  
패킷



”

---

# 네트워크를 통해 전달되는 데이터, 패킷 패킷이란?

---

“

여러 프로토콜들로  
캡슐화 된  
패킷

Ethernet	IPv4	TCP	HTTP
----------	------	-----	------

”

---

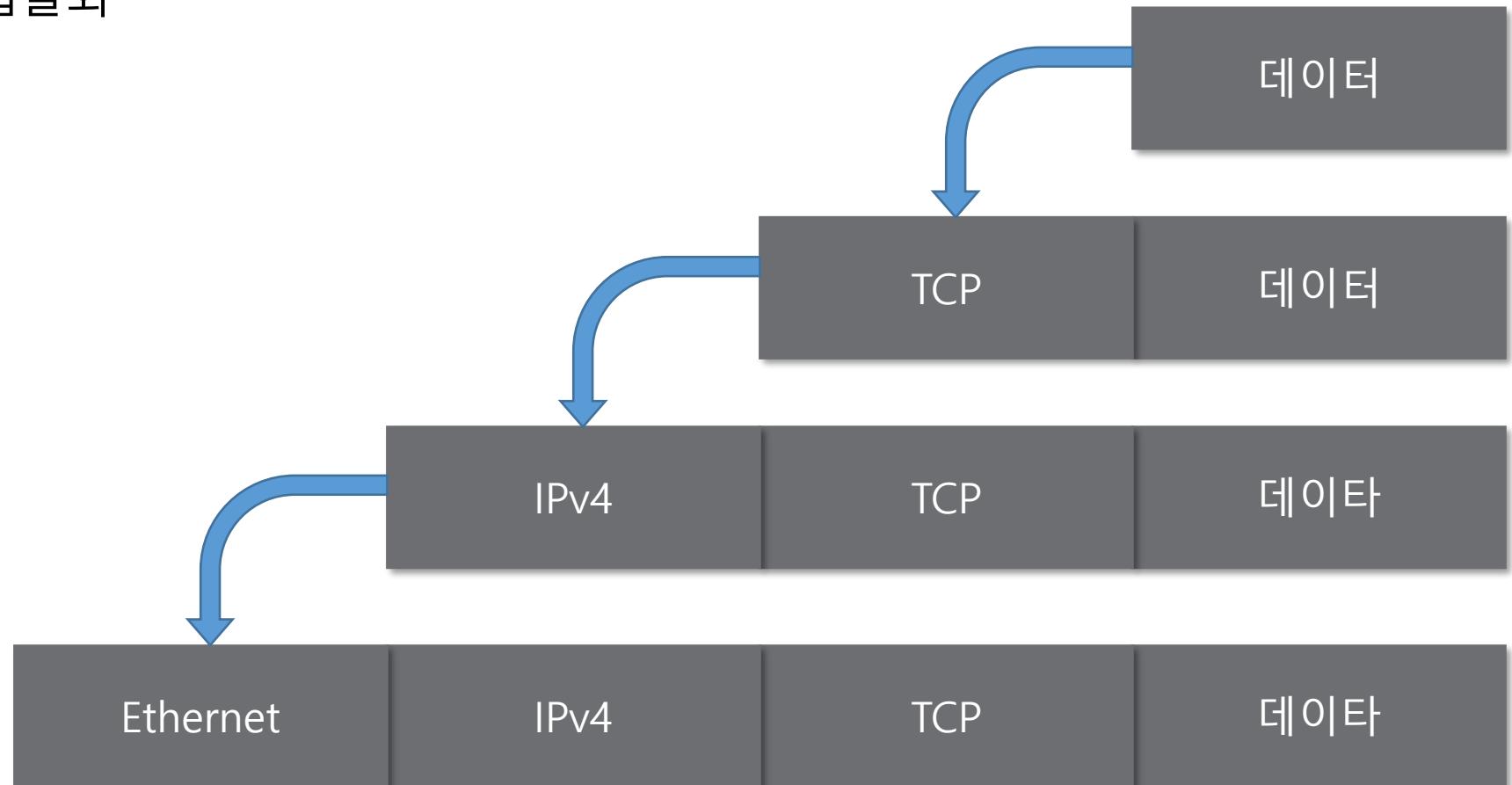
# 네트워크를 통해 전달되는 데이터, 패킷

패킷을 이용한 통신과정 - 캡슐화

“

여러 프로토콜을 이용해서  
최종적으로 [보낼 때](#)  
패킷을 만드는 과정

”



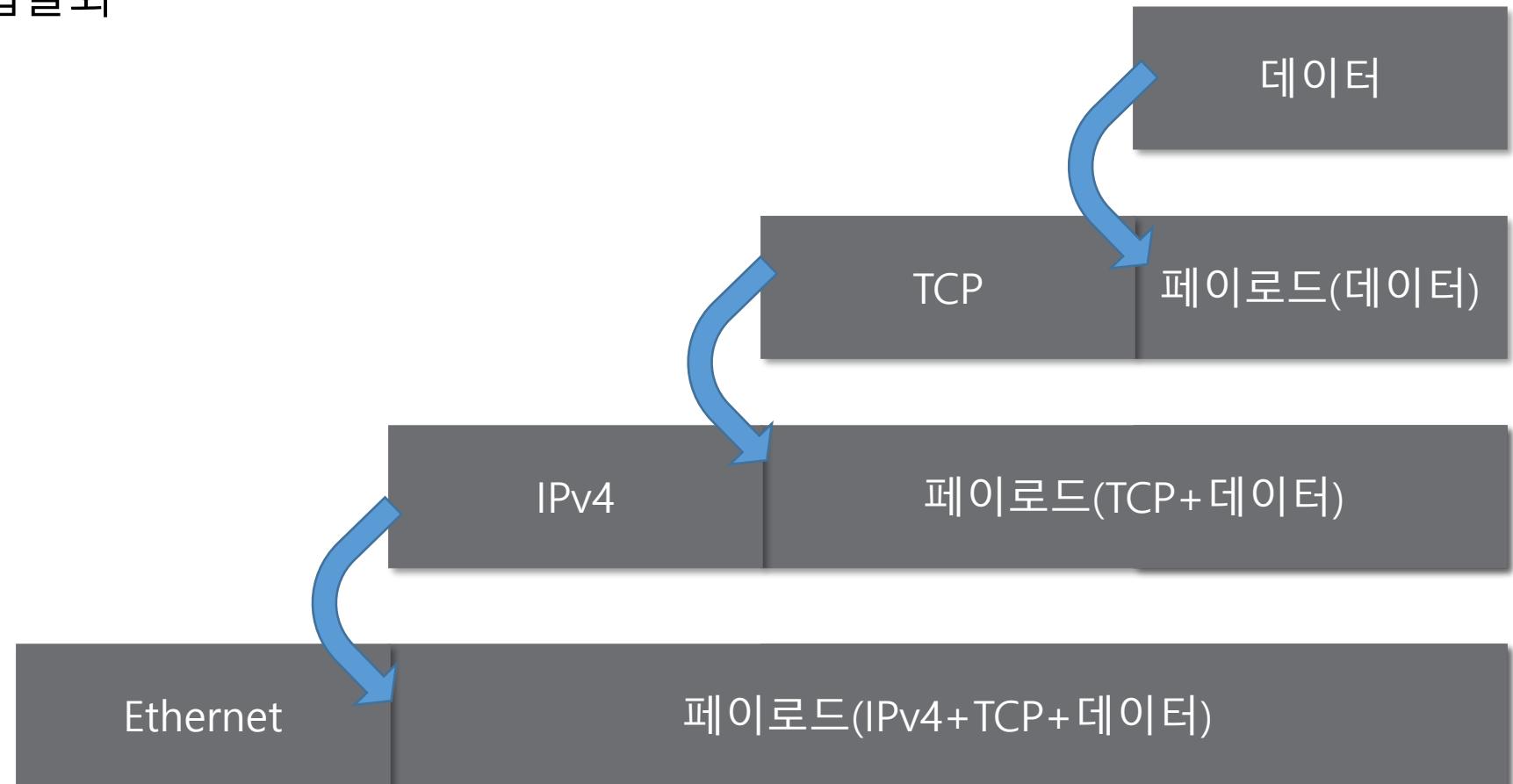
# 네트워크를 통해 전달되는 데이터, 패킷

패킷을 이용한 통신과정 - 캡슐화

“

여러 프로토콜을 이용해서  
최종적으로 보낼 때  
패킷을 만드는 과정

”



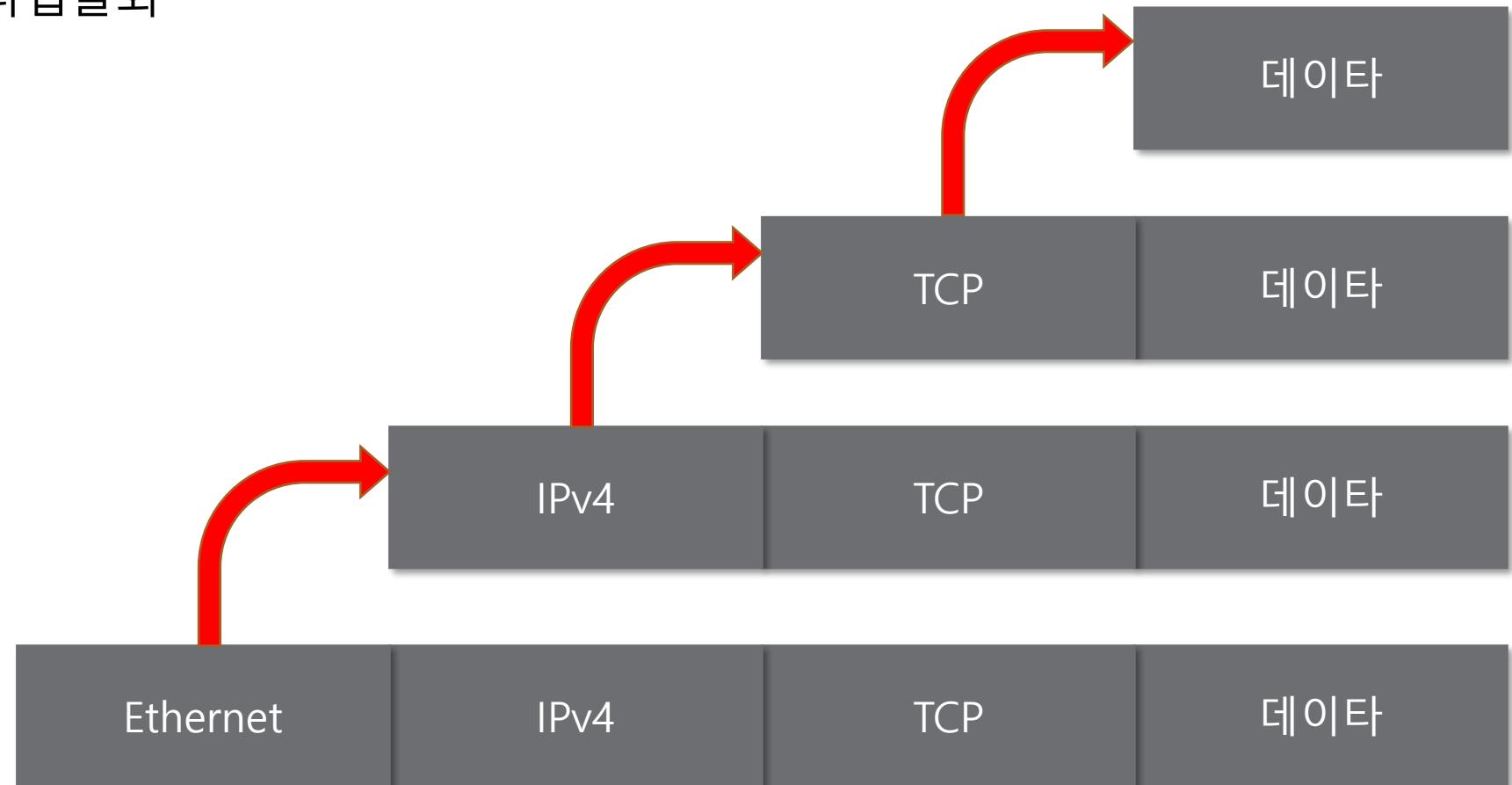
# 네트워크를 통해 전달되는 데이터, 패킷

패킷을 이용한 통신과정 - 디캡슐화

“

패킷을 받았을 때  
프로토콜들을 하나씩 확인하면서  
데이터를 확인하는 과정

“



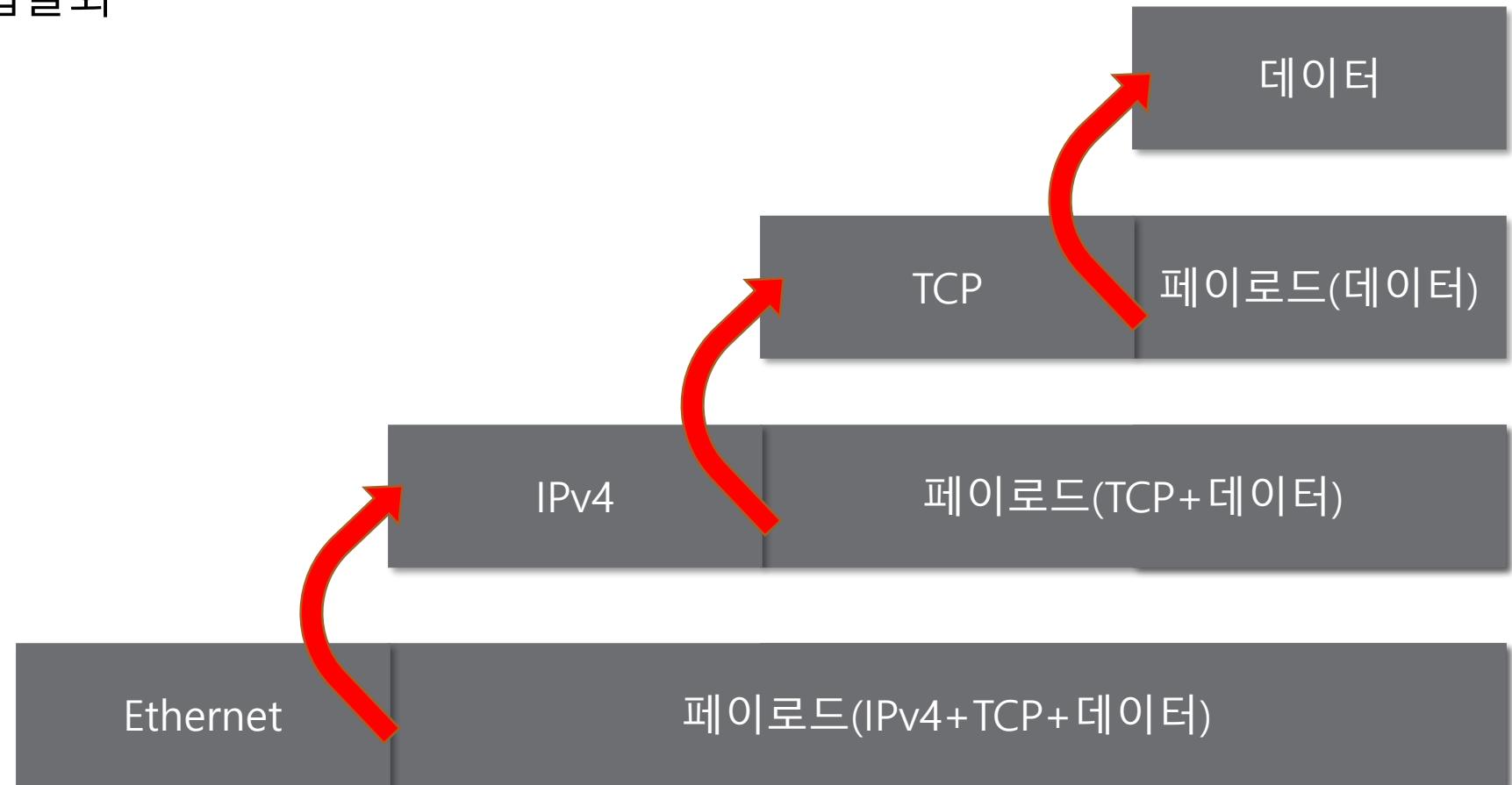
# 네트워크를 통해 전달되는 데이터, 패킷

패킷을 이용한 통신과정 - 캡슐화

“

여러 프로토콜을 이용해서  
최종적으로 받을 때  
패킷을 만드는 과정

”



# 네트워크를 통해 전달되는 데이터, 패킷

계층별 패킷의 이름 PDU

〃

계층별로 이름이 다른  
PDU

Protocol Data Unit

Ethernet

IPv4

TCP

데이터

4계층의 PDU = 세그먼트



〃

# 네트워크를 통해 전달되는 데이터, 패킷

계층별 패킷의 이름 PDU

〃

계층별로 이름이 다른  
PDU

3계층의 PDU = 패킷

Ethernet

IPv4

TCP

데이터

〃

# 네트워크를 통해 전달되는 데이터, 패킷

계층별 패킷의 이름 PDU

〃

계층별로 이름이 다른  
PDU

2계층의 PDU = 프레임

Ethernet

IPv4

TCP

데이터

〃

# 실습과제

## 1. 프로토콜의 캡슐화 된 모습과 계층별 프로토콜들을 확인해보기

Wireshark를 이용하여 패킷을 캡쳐 해보고 해당 패킷이 어떻게 캡슐화 되었는지  
자세히 살펴본다.

가까이 있는 컴퓨터끼리는 이렇게  
데이터를 주고 받는다

# 목차

## INDEX

2계층에서  
하는 일

2계층에서  
사용하는 주소

2계층  
프로토콜

실습

2계층의 기능  
2계층의 네트워크 크기

물리적인 주소  
MAC 주소

Ethernet 프로토콜

내 MAC 주소 알아보기  
Ethernet 프로토콜 캡쳐  
Ethernet 프로토콜 분석

# 2계층에서 하는 일

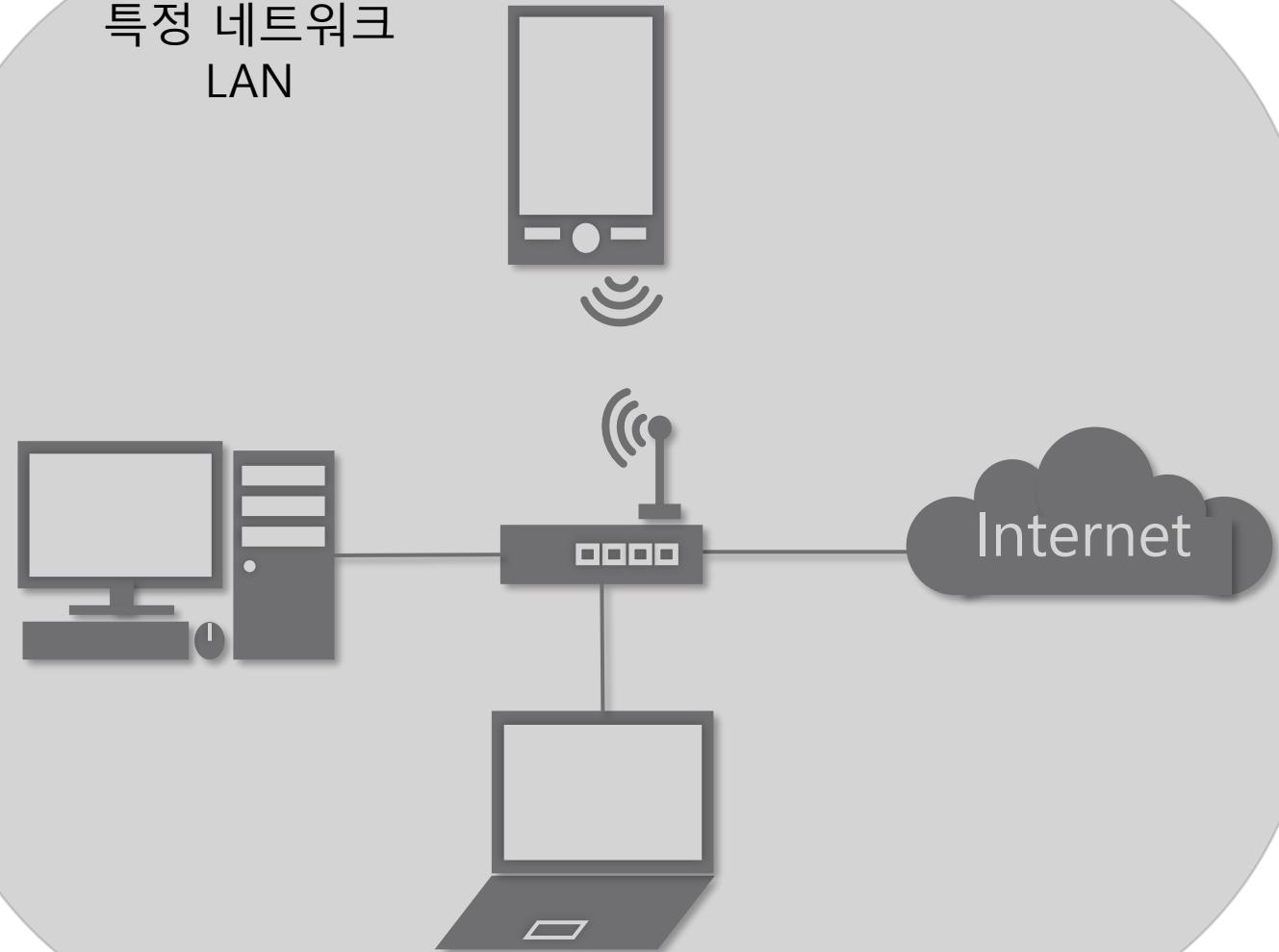
# 2계층에서 하는 일

## 2계층의 기능

2계층은 **하나의 네트워크 대역** 즉, 같은 네트워크 상에 존재하는 여러 장비들 중에서 어떤 장비가 어떤 장비에게 보내는 데이터를 전달

추가적으로 **오류제어, 흐름제어** 수행

특정 네트워크  
LAN



# 2계층에서 하는 일

## 2계층의 네트워크 크기

2계층은 **하나의 네트워크 대역 LAN**에서만 통신할 때 사용한다.

다른 네트워크와 통신할 때는 항상 **3계층**이 도와주어야 한다.

3계층의 주소와 3계층의 프로토콜을 이용하여야만 다른 네트워크와 통신이 가능하다.

특정 네트워크  
LAN



# **2계층에서 사용하는 주소**

# 2계층에서 사용하는 주소

## 물리적인 주소



LAN에서 통신할 때 사용하는  
MAC 주소

“



“

# 2계층에서 사용하는 주소

## 물리적인 주소

“

LAN에서 통신할 때 사용하는  
MAC 주소

“

선택 명령 프롬프트	
무선 LAN 어댑터 Wi-Fi:	
연결별 DNS 접미사 . . . . .	:
설명 . . . . .	: Intel(R) Dual Band Wireless-AC 7260
물리적 주소 . . . . .	: 6C-29-95-04-EB-A1
DHCP 사용 . . . . .	: 예
자동 구성 사용 . . . . .	: 예
IPv4 주소 . . . . .	: 192.168.219.100(기본 설정)
서브넷 마스크 . . . . .	: 255.255.255.0
임대 시작 날짜 . . . . .	: 2019년 3월 6일 수요일 오전 6:44:26
임대 만료 날짜 . . . . .	: 2019년 3월 9일 토요일 오후 11:58:06
기본 게이트웨이 . . . . .	: 192.168.219.1
DHCP 서버 . . . . .	: 192.168.219.1
DNS 서버 . . . . .	: 1.214.68.2 61.41.153.2
Tcpip를 통한 NetBIOS . . . . .	: 사용

# 2계층에서 사용하는 주소

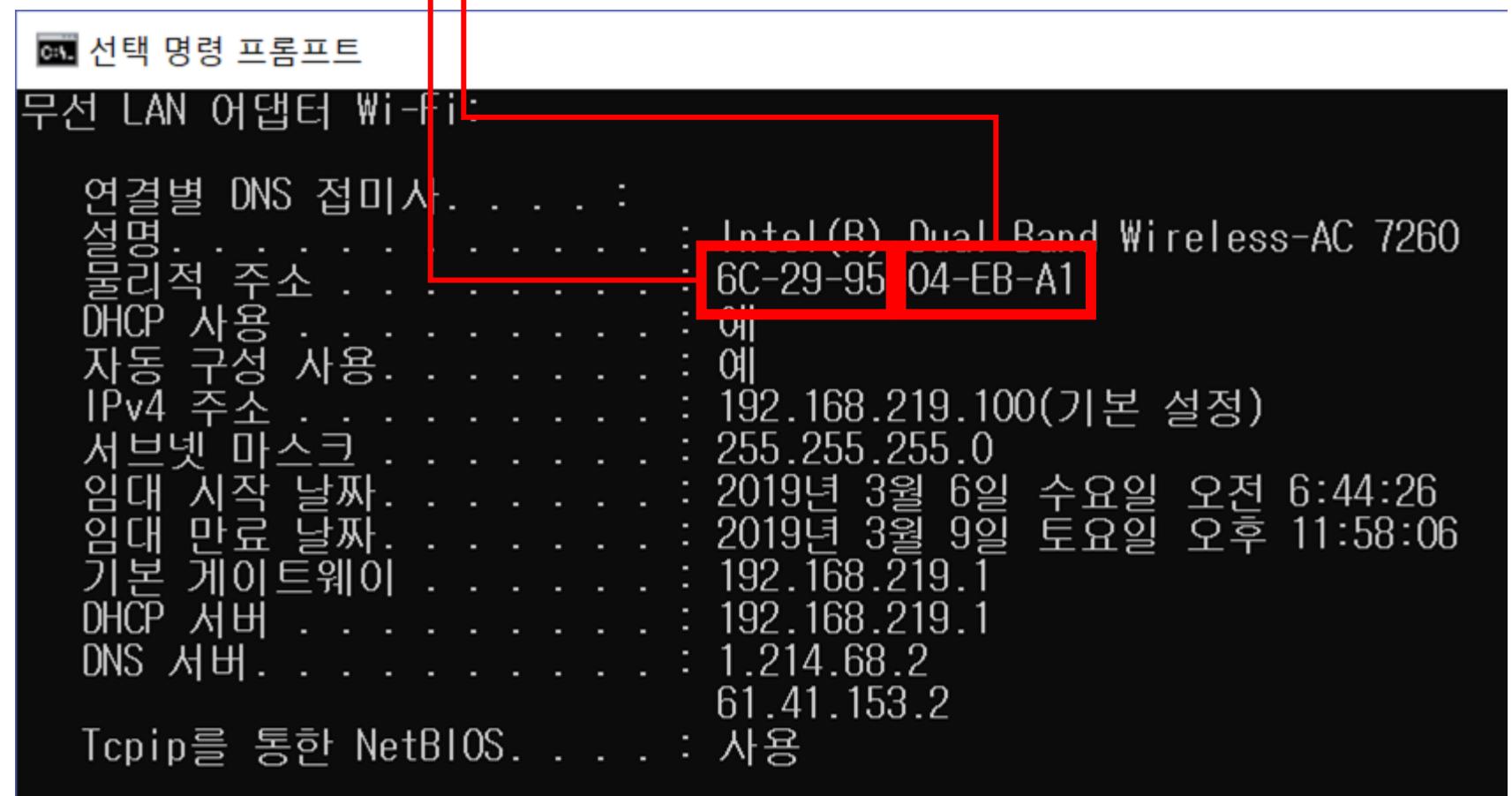
물리적인 주소

LAN에서 통신할 때 사용하는  
MAC 주소

〃

〃

- OUI : IEEE에서 부여하는 일종의 제조사 식별 ID
- 고유번호 : 제조사에서 부여한 고유번호



# **2계층 프로토콜**

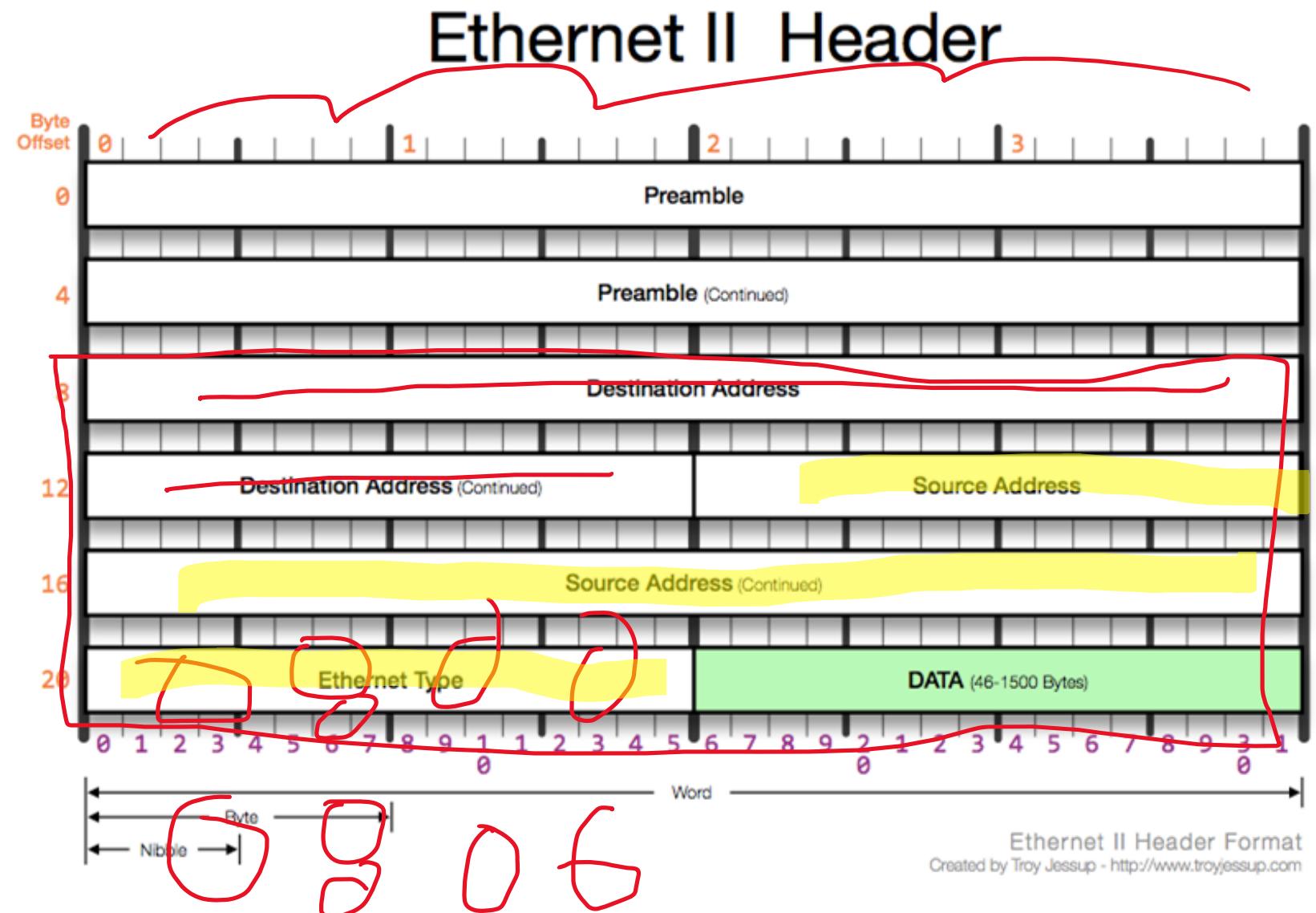
# 2계층의 프로토콜

Ethernet 프로토콜

〃

LAN에서 통신할 때 사용하는  
Ethernet 프로토콜

14 //



# 2계층의 프로토콜

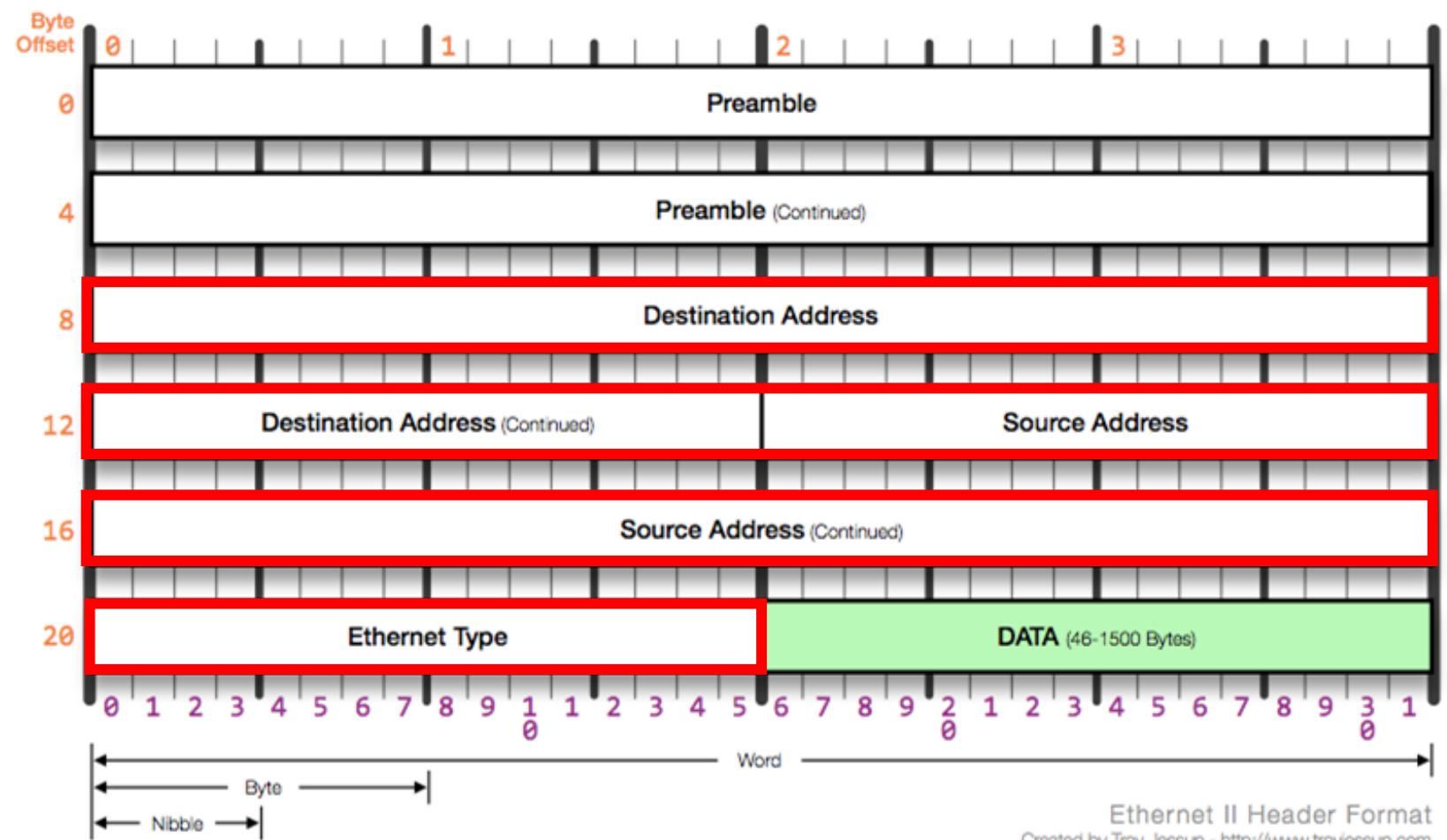
Ethernet 프로토콜

//

LAN에서 통신할 때 사용하는  
Ethernet 프로토콜

//

## Ethernet II Header



실습

## 1. 내 PC의 MAC주소 확인해보기

윈도우에서 간단하게 내PC의 MAC주소를 확인하는 방법 알아보기

## 1. Ethernet 프로토콜 캡쳐

Ethernet 프로토콜이 어떻게 생겼는지 직접 보기 위해 Wireshark를 이용해 캡쳐해보기

## 2. Ethernet 프로토콜 분석

캡쳐한 Ethernet 프로토콜에 내 MAC주소가 있는지 목적지는 어디인지 분석해보기

```
명령 프롬프트

C:\Users\oakyo>ipconfig /all

Windows IP 구성

호스트 이름 . . . . . : DESKTOP-HAA1OJO
주 DNS 접미사 . . . . . :
노드 유형 . . . . . : 혼합
IP 라우팅 사용 . . . . . : 아니요
WINS 프록시 사용 . . . . . : 아니요

이더넷 어댑터 이더넷:

미디어 상태 . . . . . : 미디어 연결 끊김
연결별 DNS 접미사 . . . . . :
설명 . . . . . : Realtek PCIe GbE Family Controller
물리적 주소 . . . . . : 90-2E-16-47-65-0F
DHCP 사용 . . . . . : 예
자동 구성 사용 . . . . . : 예

이더넷 어댑터 VirtualBox Host-Only Network:

연결별 DNS 접미사 . . . . . :
설명 . . . . . : VirtualBox Host-Only Ethernet Adapter
물리적 주소 . . . . . : 0A-00-27-00-00-08
DHCP 사용 . . . . . : 아니요
자동 구성 사용 . . . . . : 예
링크-로컬 IPv6 주소 . . . . . : fe80::a967:adb4:84ef:ac5f%8(기본 설정)
IPv4 주소 . . . . . : 192.168.238.1(기본 설정)
서브넷 마스크 . . . . . : 255.255.255.0
```

**실제로 컴퓨터끼리는  
IP주소를 사용해 데이터를 주고 받는다**

# 목차

## INDEX



### 3계층의 기능

3계층에서 하는 일  
3계층에서 쓰는 주소  
3계층 프로토콜

### 일반적인 IP 주소

Classful  
Classless  
사설IP와 공인IP

### 특수한 IP 주소

0.0.0.0  
127.X.X.X

### 실습

내 PC의 IP주소 알아보기  
네이버가 보는 내 IP주소  
알아보기

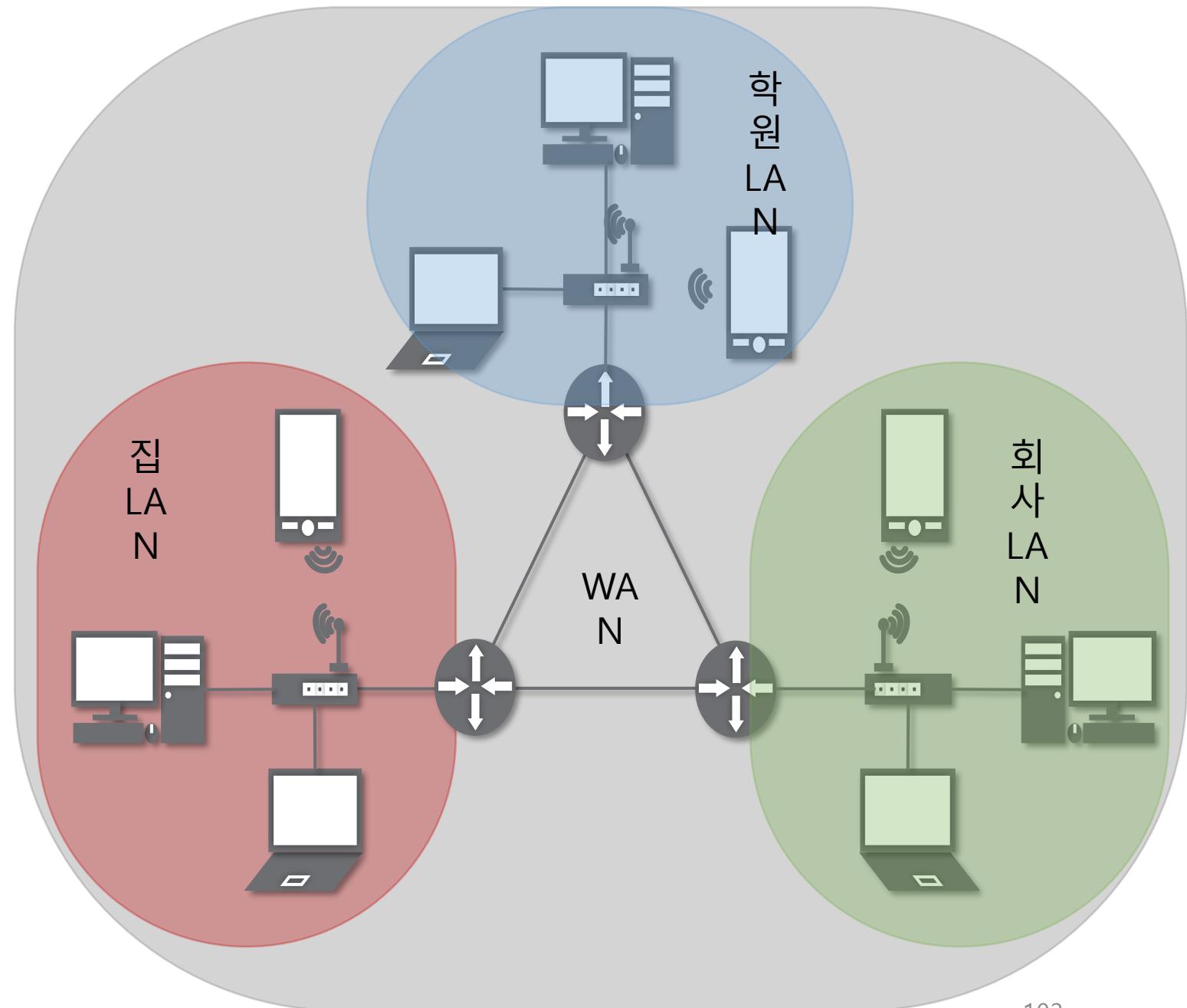
# 3계층의 기능

# 3계층의 기능

3계층에서 하는 일

3계층은 **다른 네트워크 대역**  
즉, 멀리 떨어진 곳에 존재하는  
**네트워크**까지 어떻게 데이터를  
전달할지 제어하는 일을 담당

발신에서 착신까지의 패킷의  
경로를 제어



# 3계층의 기능

## 3계층에서 쓰는 주소

“

WAN에서 통신할 때 사용하는  
IP 주소

“



# 3계층의 기능

3계층에서 쓰는 주소

//

WAN에서 통신할 때 사용하는  
IP 주소

//

```
관리자: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP 구성

이더넷 어댑터 로컬 영역 연결:

연결별 DNS 접미사. . . . . :
링크-로컬 IPv6 주소 . . . . . : fe80::10d6:6de9:e8b1:315d%12
IPv4 주소 . . . . . : 192.168.0.189
서브넷 마스크 . . . . . : 255.255.255.0
기본 게이트웨이 . . . . . : 192.168.0.1
```

# 3계층의 기능

3계층에서 쓰는 주소

〃

WAN에서 통신할 때 사용하는  
IP 주소

〃

- IPv4 주소 : 현재 PC에 할당된 IP주소
- 서브넷 마스크 : IP 주소에 대한 네트워크의 대역을 규정하는 것
- 게이트웨이 주소 : 외부와 통신할 때 사용하는 네트워크의 출입구

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP 구성

이더넷 어댑터 로컬 영역 연결:

연결별 DNS 접미사. . . . . :
링크-로컬 IPv6 주소 . . . . . : fe80::10d6:6de9:e8b1:315d%12
IPv4 주소 . . . . . : 192.168.0.189
서브넷 마스크 . . . . . : 255.255.255.0
기본 게이트웨이 . . . . . : 192.168.0.1
```

# 3계층의 기능

3계층에서 쓰는 주소

〃

WAN에서 통신할 때 사용하는  
IP 주소

〃

- IPv4 주소 : 현재 PC에 할당된 IP주소
- 서브넷 마스크 : IP 주소에 대한 네트워크의 대역을 규정하는 것
- 게이트웨이 주소 : 외부와 통신할 때 사용하는 네트워크의 출입구

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP 구성

이더넷 어댑터 로컬 영역 연결:

연결별 DNS 접미사. . . . . :
링크-로컬 IPv6 주소 . . . . . : fe80::10d6:6de9:e8b1:315d%12
IPv4 주소 . . . . . : 192.168.0.189
서브넷 마스크 . . . . . : 255.255.255.0
기본 게이트웨이 . . . . . : 192.168.0.1
```

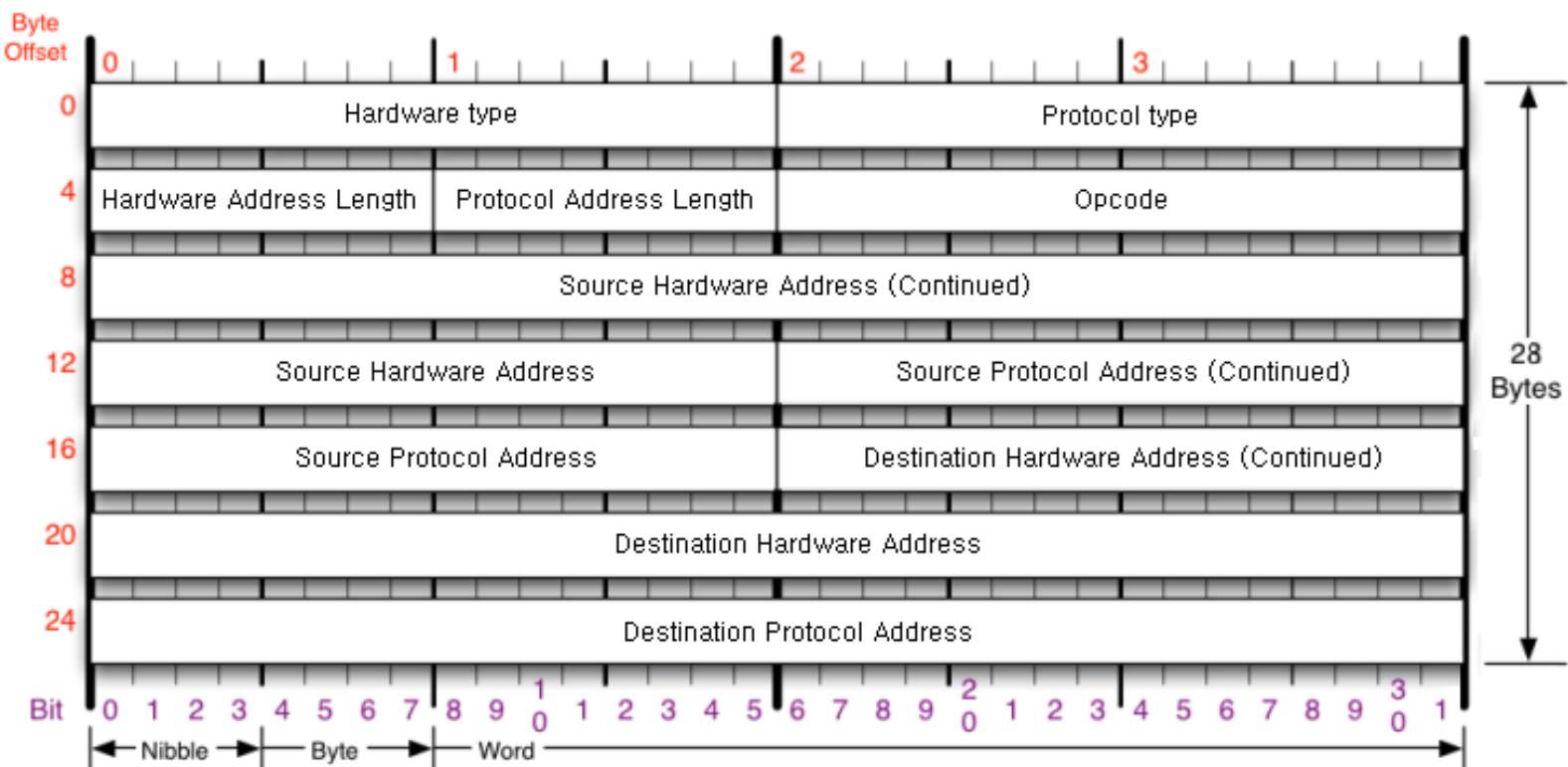
# 3계층의 기능

## 3계층 프로토콜

//

IP주소를 이용해 MAC주소를 알아오는  
ARP 프로토콜

//



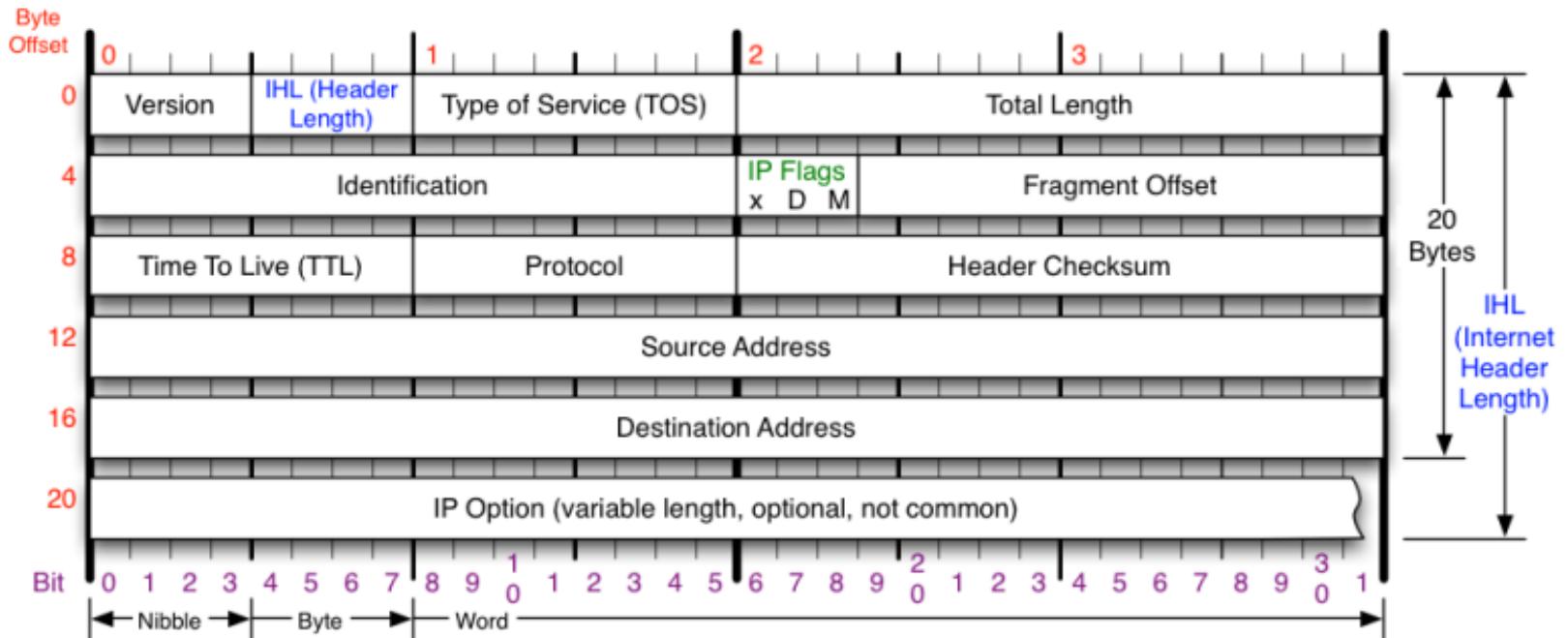
# 3계층의 기능

## 3계층 프로토콜

//

WAN에서 통신할 때 사용하는  
IPv4 프로토콜

//

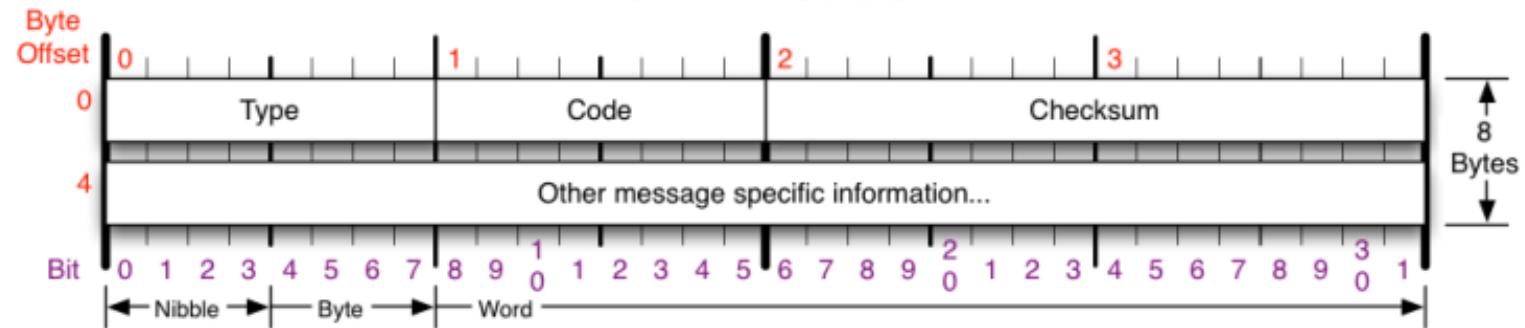


# 3계층의 기능

## 3계층 프로토콜

//

서로가 통신되는지 확인할 때 사용하는  
ICMP 프로토콜



//

# 일반적인 IP 주소

# 일반적인 IP 주소

## Classful IP 주소

낭비가 심한  
Classful IP 주소

〃

〃

A 클래스	0XXXXXXX, 첫번째 필드	0.0.0	127.255.255.255
B 클래스	10XXXXXX, 두번째 필드	128.0.0.0	191.255.255.255
C 클래스	110XXXXX, 세번째 필드	192.0.0.0	223.255.255.255
D 클래스 (멀티캐스트)	1110XXXX	224.0.0.0	239.255.255.255
E 클래스 (예약)	1111XXXX	240.0.0.0	255.255.255.255

# 일반적인 IP 주소

Classfullless IP 주소

“

낭비되지 않도록 아껴쓰는  
Classless IP 주소

”

IP주소	192.168.32.189
서브넷 마스크	255.255.255.192
네트워크 ID	192.168.32.128
브로드캐스트 주소	192.168.32.191
사용 가능 IP 범위	192.168.32.129 ~ 192.168.32.190

## IPv6 주소체계

IPv6 주소는 기존 32비트의 IPv4 주소가 고갈되는 문제를 해결하기 위하여 개발된 새로운 128비트 체계의 무제한 인터넷 프로토콜 주소를 말한다. IPv6 주소는 다음 그림과 같이 16비트 단위로 구분하며, 각 단위는 16진수로 변환되어 콜론(:)으로 구분하여 표기한다. 128비트의 IPv6 주소에서 앞의 64비트는 네트워크 주소를 의미하며, 뒤의 64비트는 네트워크에 연결된 통신장비 등에 할당되는 인터페이스 주소를 의미한다.

< IPv6 주소 표기 >

	/0~/16	~/32	~/48	~/64	~/80	~/96	~/112	~/128
16진수 표기법	0000:	0000:	0000:	0000:	0000:	0000:	0000:	0000:
기술적 경계	64비트 네트워크 주소 부문				64비트 인터페이스 주소 부문			

< IPv6 주소 장점 >

구분	주요내용
확대된 주소 공간	주소 길이가 128비트로 증가형 $2^{128}$ 개의 주소 생성 가능
단순해진 헤더 포맷	IPv4 헤더의 불필요한 필드를 제거하여 보다 빠른 처리 가능
간편해진 주소 설정기능	IPv6 프로토콜에 내장된 주소 자동 설정 기능을 이용하여 플러그 앤 플레이 설치가 가능
강화된 보안 기능	IPv6에서는 IPSec 기능을 기본 사항으로 제공
개선된 모바일 IP	IPv6 헤더에서 이동성 지원

### ※참고 **Prefix**

**Prefix** 표기란 서브넷 마스크 맨 앞의 비트부터 1의 개수를 표기하는 방식을 말한다. 즉 맨 앞에 비트부터 공통 비트 개수를 표기하는 방식이다. 예를 들어 서브넷 마스크가 255.255.255.0인 경우 맨 앞의 비트부터 1이 24개가 있으므로 /24로 표기한다.

### IP 주소 클래스 (IP Address Class)

IP 주소 범위는 0.0.0.0 ~ 255.255.255.255까지 포함될 수 있다. 하지만 5개의 클래스로 정의됨으로써 IP 주소 낭비 방지와 효율적인 서브넷 관리가 가능하다.

#### A Class (Unicast Address)

첫 번째 필드를 2진수로 변환할 경우 맨 앞에 0이라는 공통 비트를 갖는다면 A Class로 정의한다. (00000000 ~ 01111111 즉 0부터 127까지) A Class가 사용하는 기본 서브넷 마스크는 255.0.0.0이며 호스트 아이디가 24bit이므로 네트워크 아이디당 나올 수 있는 IP 주소 개수는  $2^{24}$  즉 16,777,216개가 된다.

A Class IP Address Range & Default Subnet Mask							
1	8	9	16	17	24	25	32
0NNNNNNN	HHHHHHHH	HHHHHHHH	HHHHHHHH	HHHHHHHH			
Network				Host			
<b>Range</b>							
0 00000000	0 00000000	0 00000000	0 00000000		0.0.0.0		
0 11111111	1 11111111	1 11111111	1 11111111		~		
0 11111111	1 11111111	1 11111111	1 11111111		127.255.255.255		
<b>Default Subnet Mask</b>							
11111111	00000000	00000000	00000000		255.0.0.0		

### B Class (Unicast Address)

첫 번째 필드를 2진수로 변환할 경우 맨 앞에 10이라는 공통 비트를 갖는다면 B Class로 정의한다. (10000000 ~ 10111111 즉 128부터 191까지) B Class가 사용하는 기본 서브넷 마스크는 255.255.0.0이며 호스트 아이디가 16bit이므로 네트워크 아이디당 나올 수 있는 IP 주소 개수는  $2^{16}$ 개 즉 65,536개가 된다.

B Class IP Address Range & Default Subnet Mask									
1	8	9	16	17	24	25	32		
10NNNNNN	NNNNNNNN	HHHHHHHH	HHHHHHHH						
Network				Host					
<b>Range</b>									
<b>10000000</b>   00000000   00000000   00000000				128.0.0.0					
<b>10111111</b>   11111111   11111111   11111111				~ 191.255.255.255					
<b>Default Subnet Mask</b>									
<b>11111111</b>   11111111   00000000   00000000				255.255.0.0					

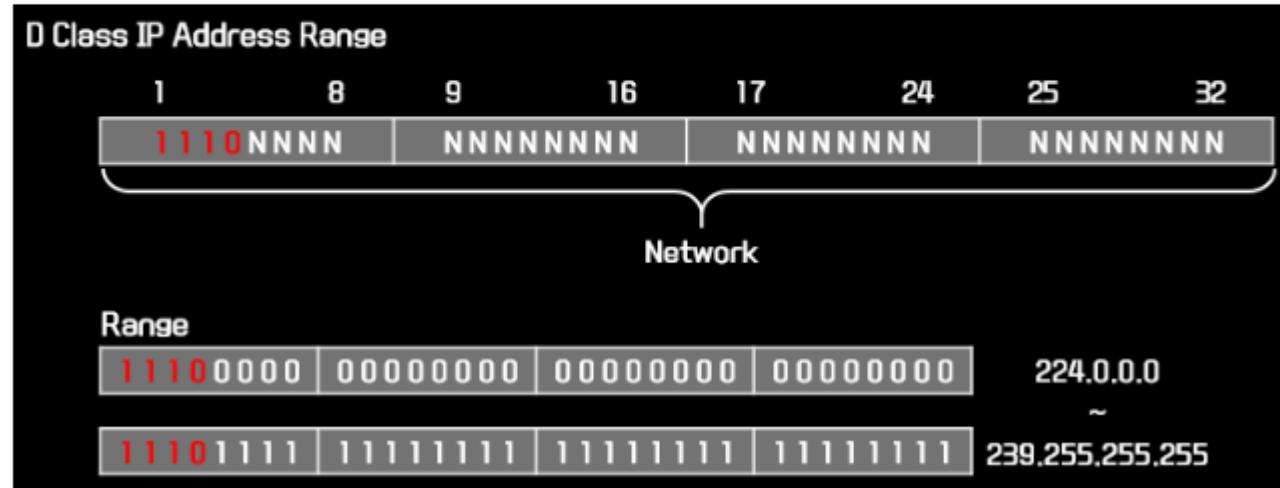
### C Class (Unicast Address)

첫 번째 필드를 2진수로 변환할 경우 맨 앞에 110이라는 공통 비트를 갖는다면 C Class로 정의한다. (11000000 ~ 11011111 즉 192부터 223까지) C Class가 사용하는 기본 서브넷 마스크는 255.255.255.0이며 호스트 아이디가 8bit이므로 네트워크 아이디당 나올 수 있는 IP 주소 개수는  $2^8$ 개 즉 256개가 된다.

C Class IP Address Range & Default Subnet Mask									
1	8	9	16	17	24	25	32		
110NNNNN	NNNNNNNN	NNNNNNNN	NNNNNNNN	HHHHHHHH					
Network				Host					
<b>Range</b>									
11000000	00000000	00000000	00000000	192.0.0.0		~			
11011111	11111111	11111111	11111111	223.255.255.255					
<b>Default Subnet Mask</b>									
11111111	11111111	11111111	00000000	255.255.255.0					

### D Class (Multicast Address)

첫 번째 필드를 2진수로 변환할 경우 맨 앞에 1110이라는 공통 비트를 갖는다면 D Class로 정의한다. (11100000 ~ 11101111 즉 224부터 239까지) D Class는 멀티캐스트 주소로 예약되어 있으며 서브넷 마스크를 이용하여 블락 단위로 동작하지 않기 때문에 서브넷 마스크를 사용하지 않는다.



### E Class (Broadcast Address)

첫 번째 필드를 2진수로 변환할 경우 맨 앞에 1111이라는 공통 비트를 갖는다면 E Class로 정의한다. (11110000 ~ 11111111 즉 240부터 255까지) E Class는 IANA에서 사용을 제한 시킨 주소이기 때문에 네트워크 인터페이스에 설정이 불가능하다.

E Class IP Address Range								* R : 예약을
1	8	9	16	17	24	25	32	
1111RRRR		Reserved		Reserved		Reserved		
Network								
Range								
11110000	00000000	00000000	00000000		240.0.0.0			
11111111	11111111	11111111	11111111		~			
					255.255.255.255			

## 네트워크 이름과 서브넷 브로드캐스트 주소

## 네트워크 이름

네트워크 이름은 네트워크 아이디의 호스트 부분이 모두 0인 주소를 말한다. 이 주소는 네트워크를 표기하거나 라우팅 경로로 사용하기 위해서 예약된 값이기 때문에 인터페이스에 설정이 불가능하다.

예를 들어 192.168.1.1/24라는 주소의 네트워크 아이디는 192.168.1이며 뒤에 8bit는 호스트 아이디이다. 이 때 호스트 아이디 전체가 0인 주소 즉 192.168.1.0을 네트워크 이름이라고 한다.

서브넷 브로드캐스트 주소

서브넷 브로드캐스트 주소는 네트워크 아이디의 호스트 부분이 모두 1인 주소를 말한다. 이 주소는 네트워크에서 브로드캐스트를 실시할 때 예약된 값이기 때문에 인터페이스에 설정이 불가능하다.

예를 들어 192.168.1.1/24라는 주소의 네트워크 아이디는 192.168.1이며 뒤에 8bit는 호스트 아이디이다. 이 때 호스트 아이디 전체가 1인 주소 즉 192.168.1.255를 서브넷 브로드캐스트 주소라고 한다.

따라서 인터페이스에 설정 가능한 IP 주소는 서브넷 마스크의 호스트 아이디에 의해서 정해지는데 이 때 사용하지 못하는 네트워크 아이디와 서브넷 브로드캐스트 주소 2개를 뺀 나머지가 사용 가능한 IP 주소의 갯수가 된다.

예를 들어 192.168.1.1/24의 IP 주소는 총 256개이다. (192.168.1.0 ~ 192.168.1.255) 하지만 이 중 네트워크 아이디(192.168.1.0)와 서브넷 브로드캐스트 주소(192.168.1.255)를 제외한 나머지 254개(192.168.1.1 ~ 192.168.1.254)의 IP가 할당 가능한 IP 주소의 갯수가 된다.

아래의 예를 보자

-192.168.1.0/24

.주소 범위 : 192.168.1.0 ~ 192.168.1.255

.Network-ID : 192.168.1.**00000000** = 192.168.1.0

.Broadcast 주소 : 192.168.1.**11111111** = 192.168.1.255

.사용 가능 주소 : 192.168.1.1 ~ 192.168.1.254

-172.16.0.0/16

.주소 범위 : 172.16.0.0 ~ 172.16.255.255

.Network-ID : 172.16.**00000000.00000000** = 172.16.0.0

.Broadcast 주소 : 172.16.**11111111.11111111** = 172.16.255.255

.사용 가능 주소 : 172.16.0.1 ~ 172.16.255.254

-10.0.0.0/8

.주소 범위 : 10.0.0.0 ~ 10.255.255.255

.Network-ID : 10.**00000000.00000000.00000000** = 10.0.0.0

.Broadcast 주소 : 10.**11111111.11111111.11111111** = 10.255.255.255

.사용 가능 주소 : 10.0.0.1 ~ 10.255.255.254

## 공인 IP 주소와 사설 IP 주소

### 공인 IP 주소(Public IP Address)

공인 IP 주소는 인터넷과 같은 공인 환경에 직접 연결이 가능한 주소를 말한다. 즉 인터넷을 하기 위해서는 공인 IP 주소가 필요하며 이 주소는 ISP 업체로부터 임대를 받아서 사용해야 한다.

### 사설 IP 주소(Private IP Address)

공인 환경이 아닌 기업 내부 사설 환경에서 사용을 권장하는 주소이다. 이 주소는 인터넷과 연결되지 않기 때문에(WAN 구간을 연결하는 Router는 사설 IP 주소를 외부로 전송할 수 없다) 다른 환경과 중첩하여 주소 범위를 사용해도 무관하며 ISP 업체로부터 임대를 받지 않아도 사용할 수 있다.

#### 사설 IP 주소 범위

Class	Address Range	Prefix
A Class	10.0.0.0 ~ 10.255.255.255	10.0.0.0/8
B Class	172.16.0.0 ~ 172.31.255.255	172.16.0.0/12
C Class	192.168.0.0 ~ 192.168.255.255	192.168.0.0/16

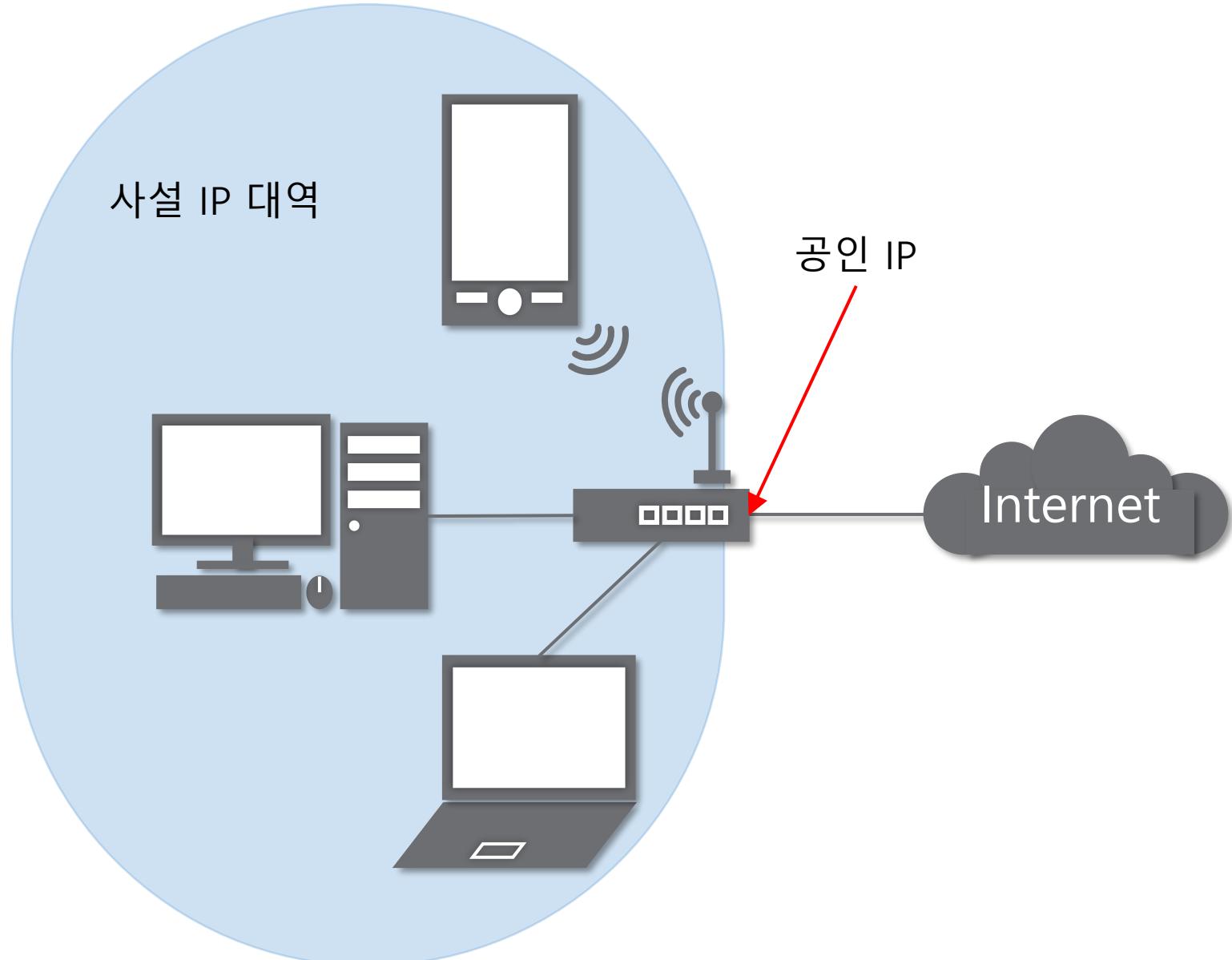
# 일반적인 IP 주소

사설 IP와 공인 IP

〃

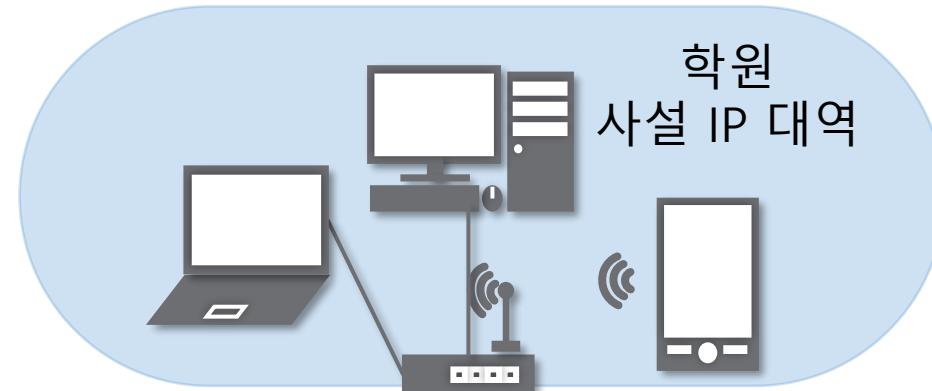
공인IP 1개당  $2^{32}$ 개의 사설IP  
사설IP와 공인IP

〃



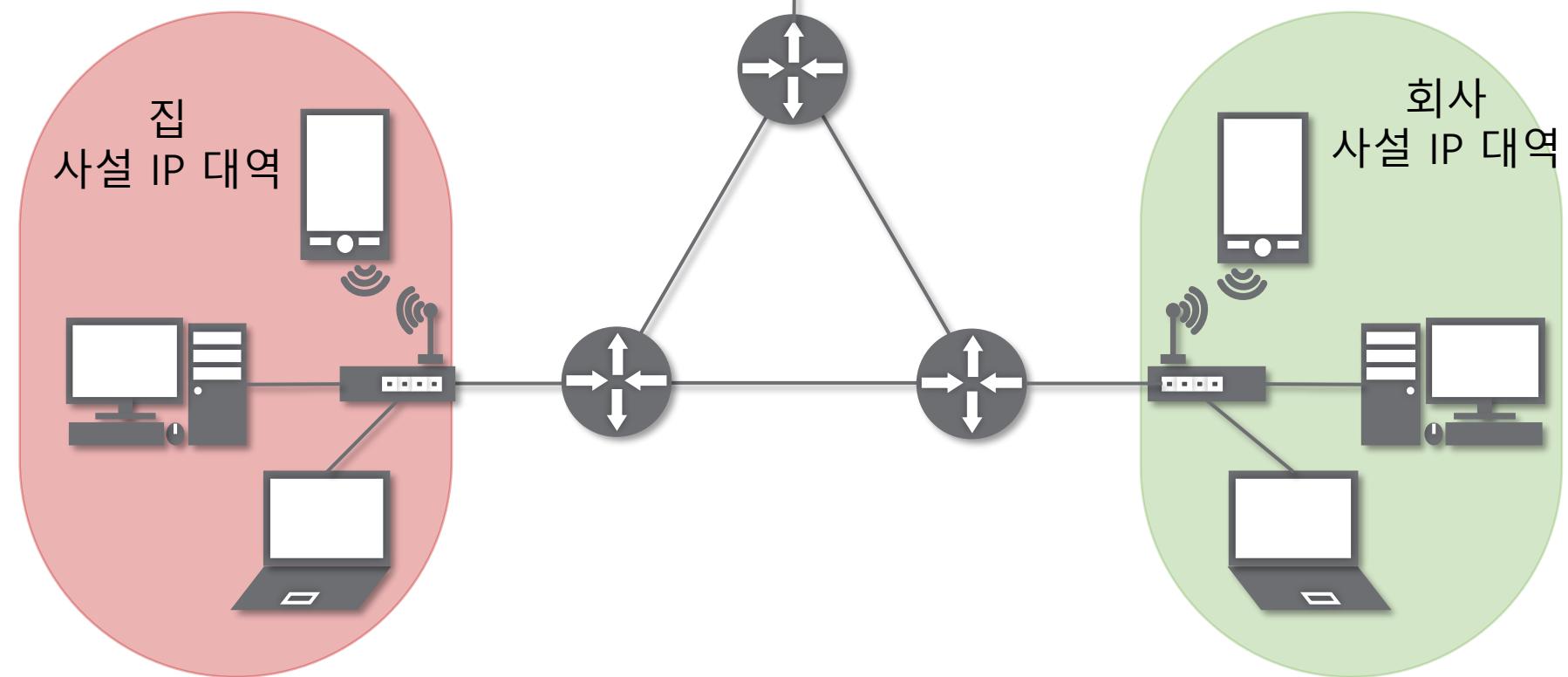
# 일반적인 IP 주소

사설 IP와 공인 IP



실제 일반적인 네트워크의 모습  
사설IP와 공인IP

〃



〃

# 일반적인 IP 주소

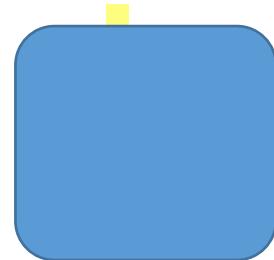
사설 IP와 공인 IP

〃

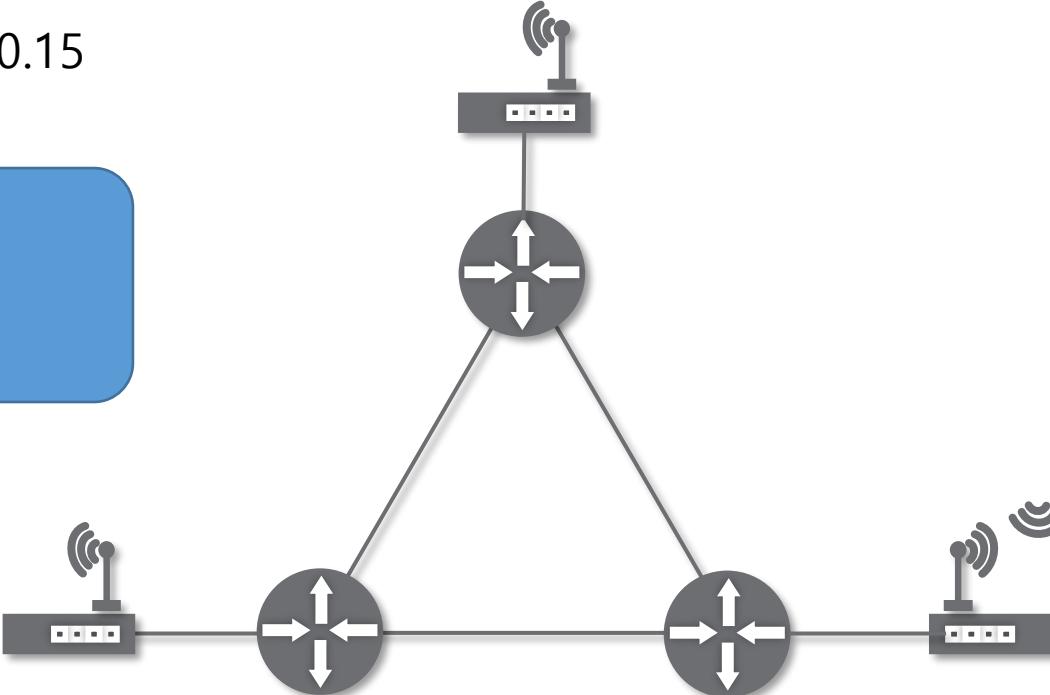
인터넷 세상에서 바라본 모습  
사설IP와 공인IP

〃

168.192.0.15



222.237.190.176



실제 인터넷 세상에서는 공인 IP로만 통신  
외부 네트워크 대역에서는 사설IP 대역이 보이지 않는다.

**특수한 IP 주소**

# 특수한 IP 주소

0.0.0.0/0

---

〃

Wildcard  
0.0.0.0

〃

---

IPv4 경로 테이블			
활성 경로:			
네트워크 대상	네트워크 마스크	게이트웨이	
0.0.0.0	0.0.0.0	192.168.0.1	연결됨
127.0.0.0	255.0.0.0		연결됨
127.0.0.1	255.255.255.255		연결됨
127.255.255.255	255.255.255.255		연결됨
192.168.0.0	255.255.255.0		연결됨
192.168.0.189	255.255.255.255		연결됨
192.168.0.255	255.255.255.255		연결됨

# 특수한 IP 주소

127.0.0.1

---

//

나 자신을 나타내는 주소

127.0.0.1

---

//

```
C:\ 관리자: C:\Windows\system32\cmd.exe
C:\Users\Administrator>ping 127.0.0.1

Ping 127.0.0.1 32바이트 데이터 사용:
127.0.0.1의 응답: 바이트=32 시간<1ms TTL=128

127.0.0.1에 대한 Ping 통계:
    패킷: 보냄 = 4, 받음 = 4, 손실 = 0 (0% 손실),
    왕복 시간(밀리초):
        최소 = 0ms, 최대 = 0ms, 평균 = 0ms
```

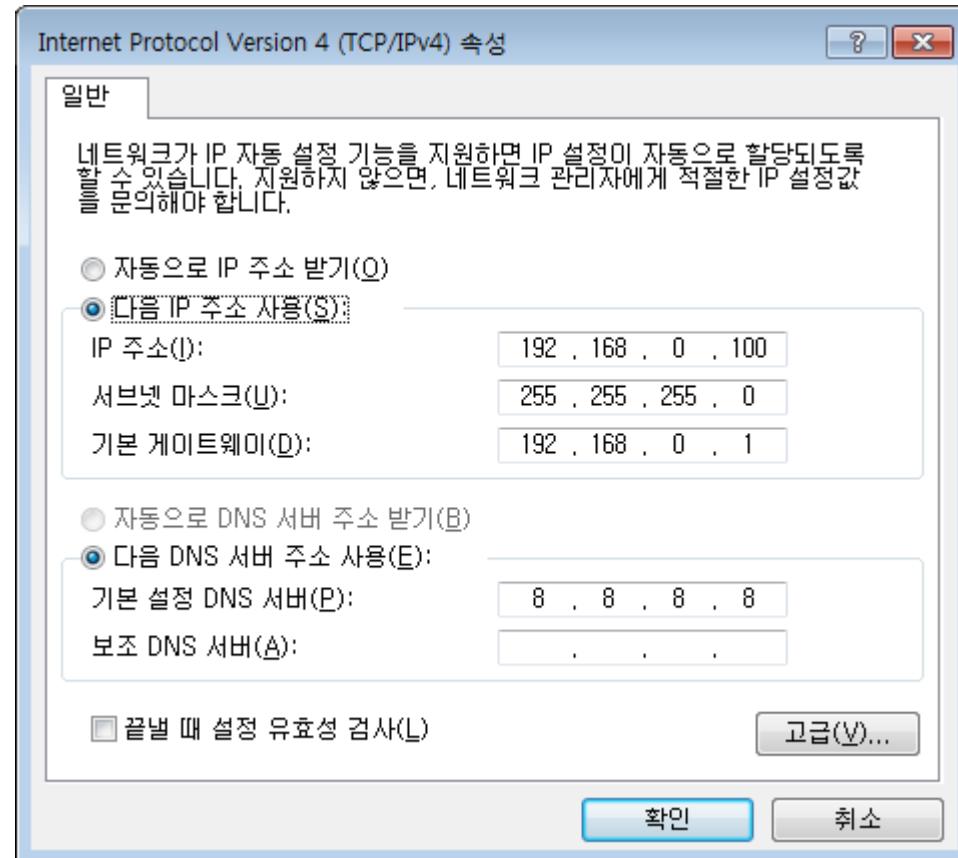
# 특수한 IP 주소

## 게이트웨이 주소

어딘가로 가려면 일단 여기로  
게이트웨이 주소

//

//



실습

## 1. 내 PC의 IP주소 확인해보기

윈도우에서 간단하게 내PC의 IP주소를 확인하는 방법 알아보기

## 1. 네이버 서버가 알고 있는 나의 IP주소 확인해보기

네이버 서버와 통신할 때 네이버 서버가 알고 있는 나의 IP주소를 알아보고 1.에서 확인한 IP와 비교해보기

# **통신하기 전 반드시 필요한 ARP 프로토콜**

# 목차

## INDEX

### ARP 프로토콜

### ARP 프로토콜의 통신 과정

### ARP 테이블

### 실습

ARP가 하는 일  
ARP 프로토콜의 구조

IP주소로 MAC주소를  
알아오는 과정

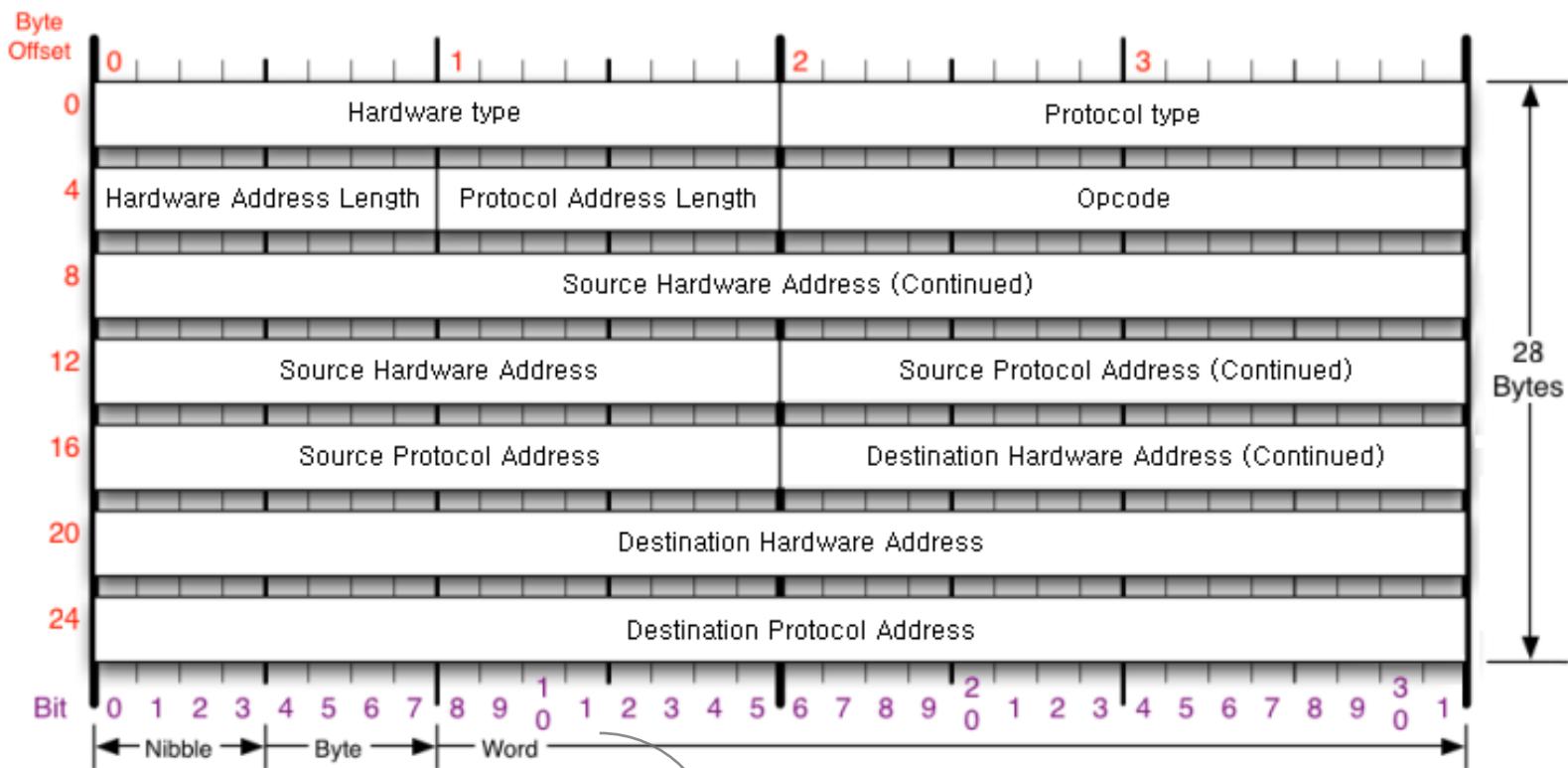
나와 통신했던  
컴퓨터들

ARP 테이블 확인해보기  
ARP 프로토콜 분석하기

# **ARP 프로토콜**

# ARP 프로토콜

ARP가 하는 일



ARP 프로토콜은 같은 네트워크 대역에서 통신을 하기 위해 필요한 MAC주소를 IP주소를 이용해서 알아오는 프로토콜이다.

같은 네트워크 대역에서 통신을 한다고 하더라고 데이터를 보내기 위해서는 7계층부터 캡슐화를 통해 데이터를 보내기 때문에 IP주소와 MAC주소가 모두 필요하다. 이 때 IP주소는 알고 MAC 주소는 모르더라고 ARP를 통해 통신이 가능하다.

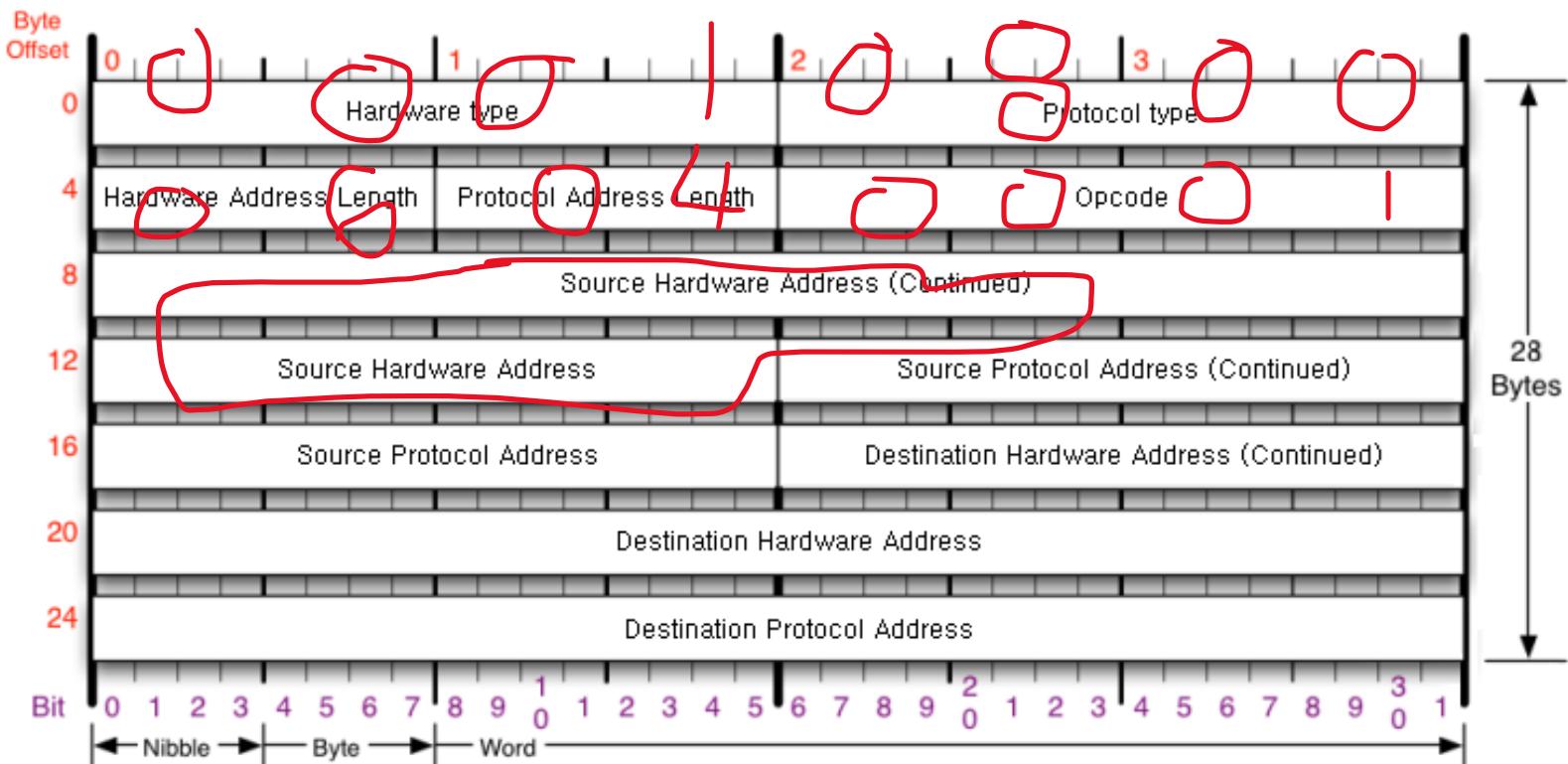
# ARP 프로토콜

## ARP 프로토콜의 구조

//

IP주소를 이용해 MAC주소를 알아오는  
ARP 프로토콜

//



# **ARP 프로토콜의 통신 과정**

# ARP 프로토콜의 통신 과정

IP 주소로 MAC 주소를 알아오는 과정

---

〃

IP주소만 알고 있을 때?  
ARP로 MAC주소를  
알아오기

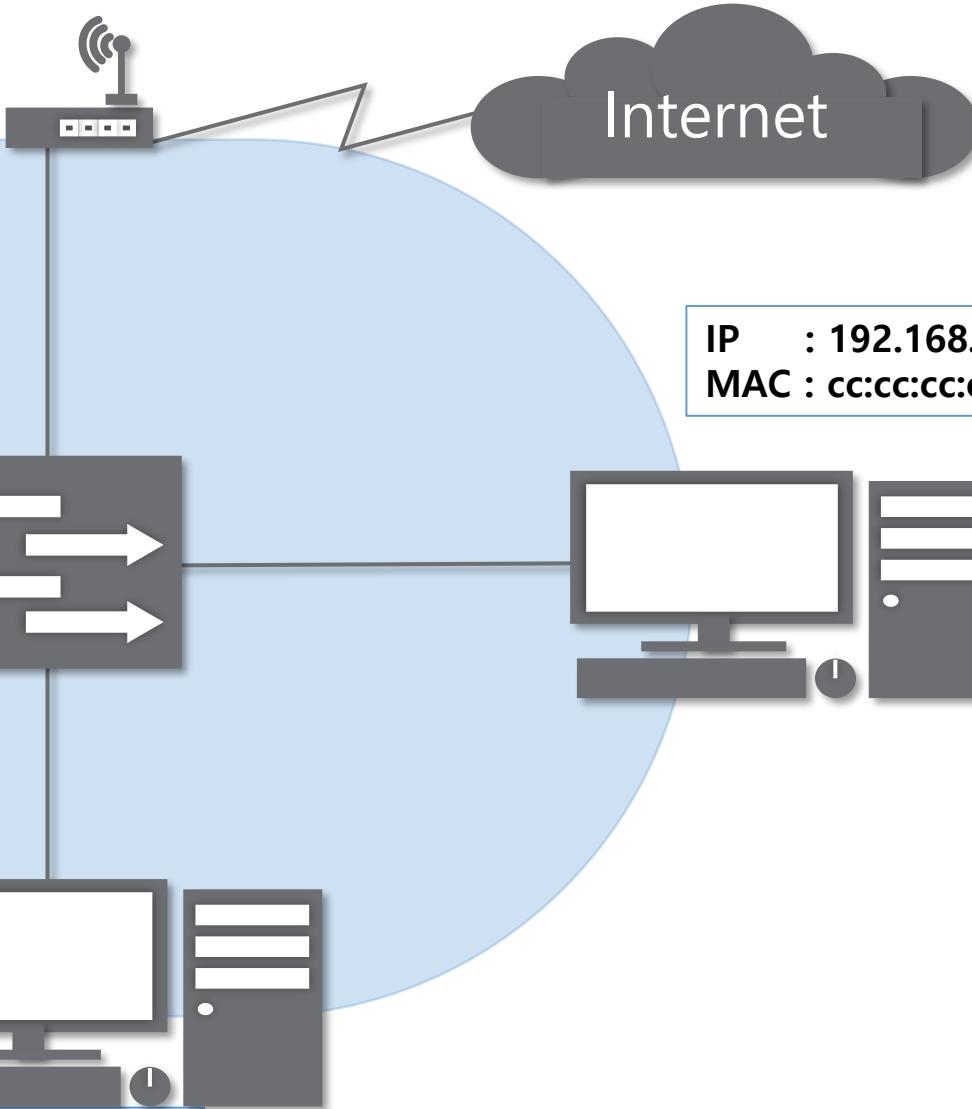
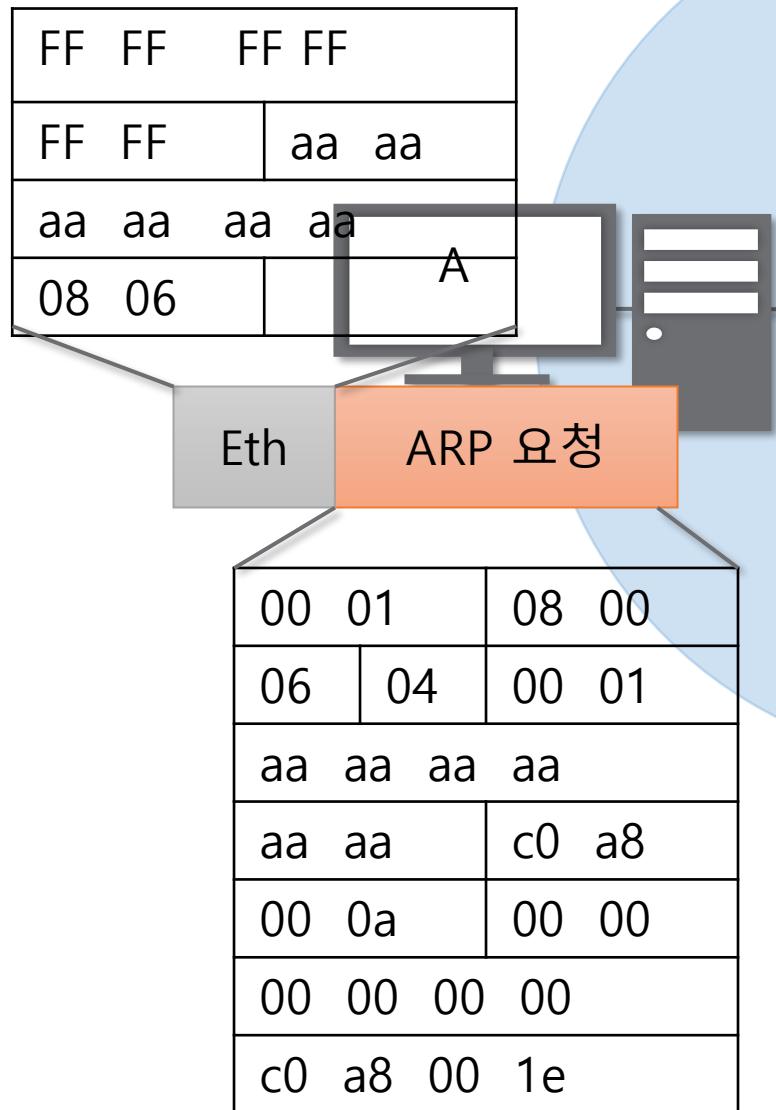
〃

---

IP : 192.168.0.10  
MAC : aa:aa:aa:aa:aa:aa

IP : 192.168.0.40  
MAC : dd:dd:dd:dd:dd:dd

Internet

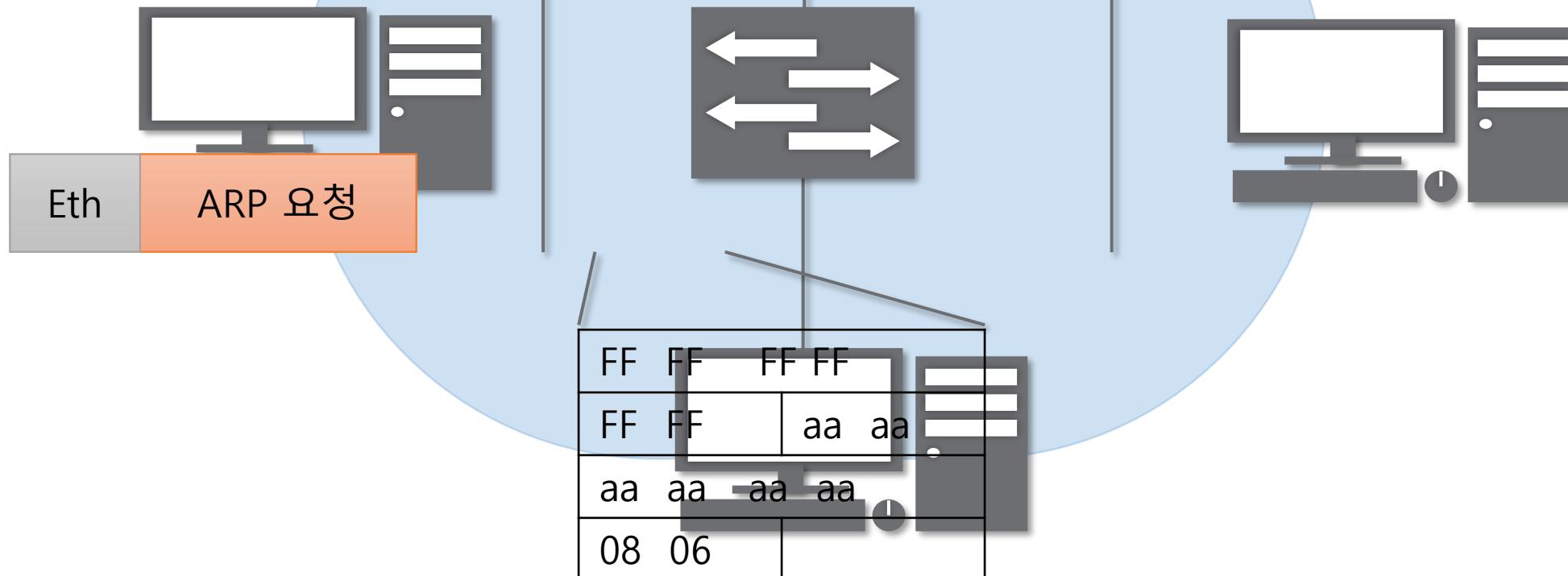


IP : 192.168.0.40  
MAC : dd:dd:dd:dd:dd:dd

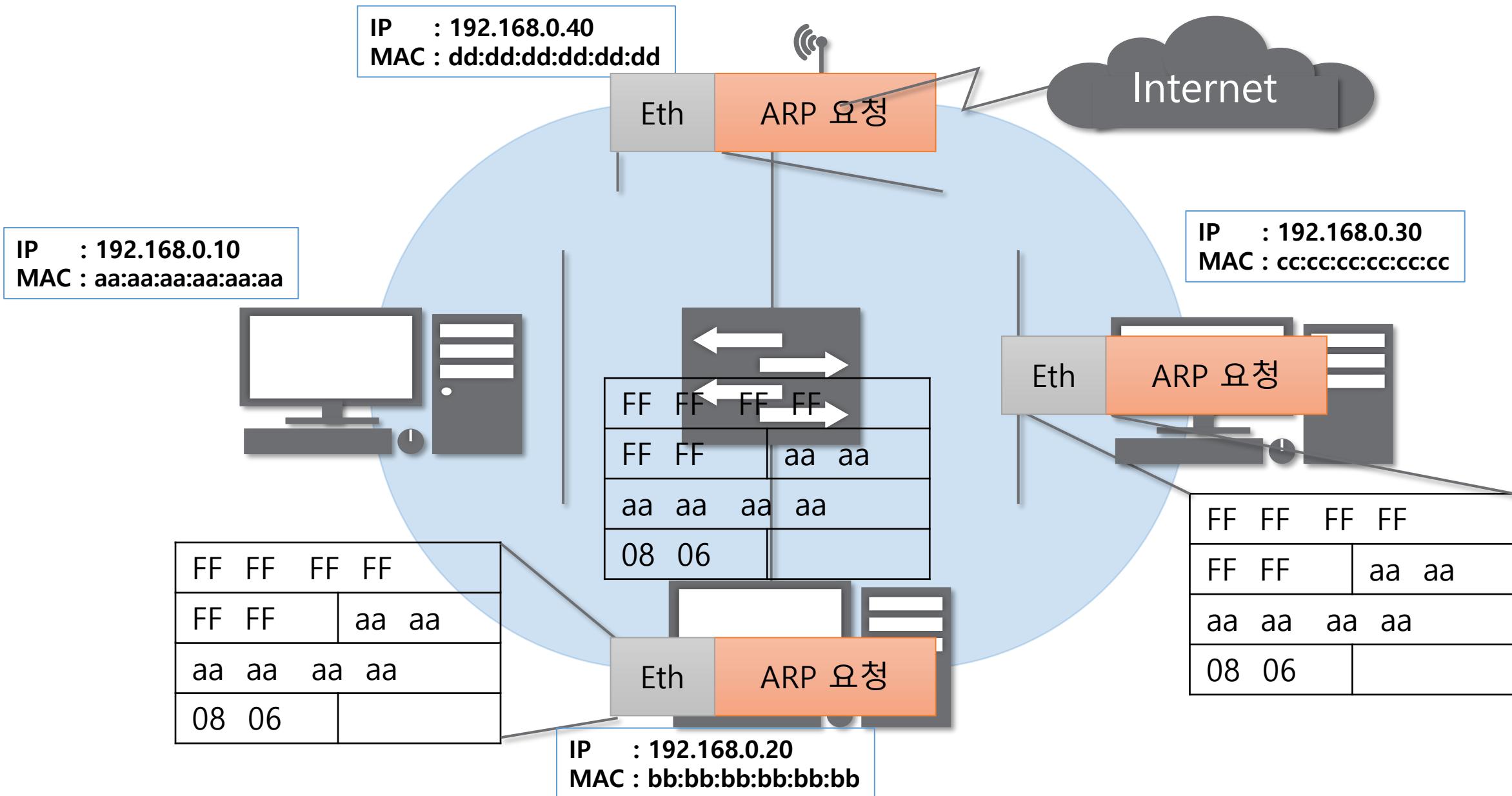
Internet

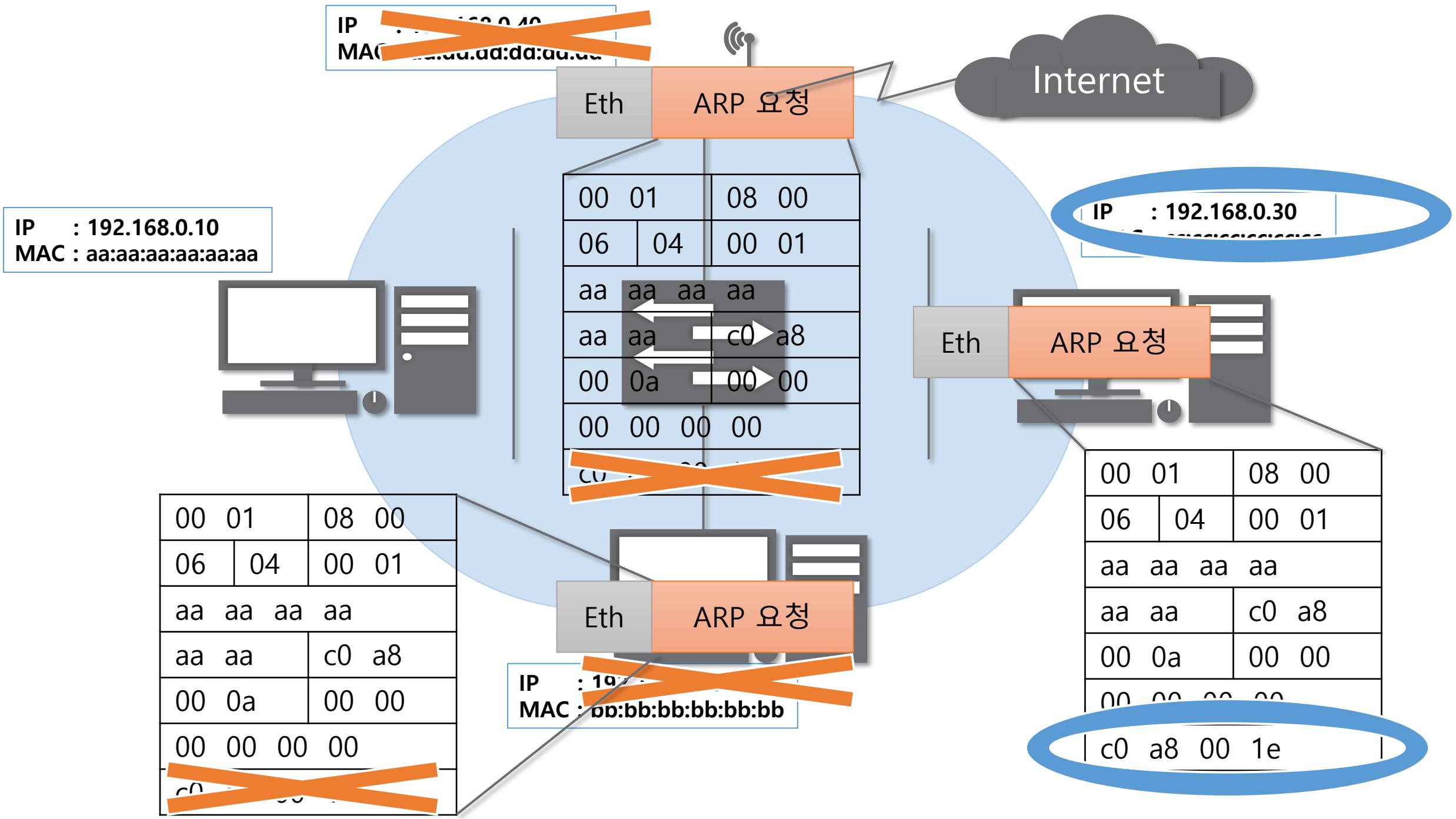
IP : 192.168.0.10  
MAC : aa:aa:aa:aa:aa:aa

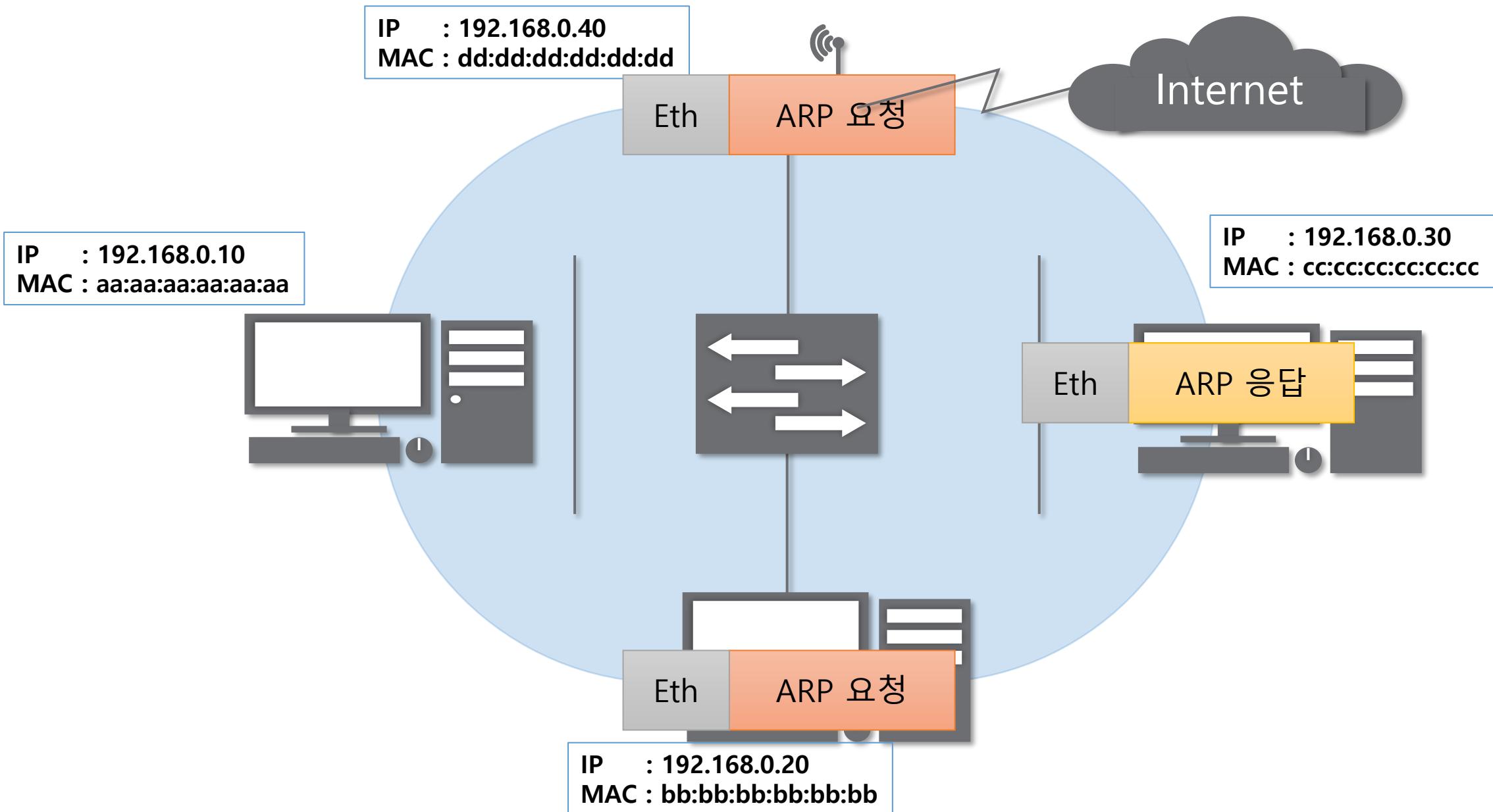
IP : 192.168.0.30  
MAC : cc:cc:cc:cc:cc:cc

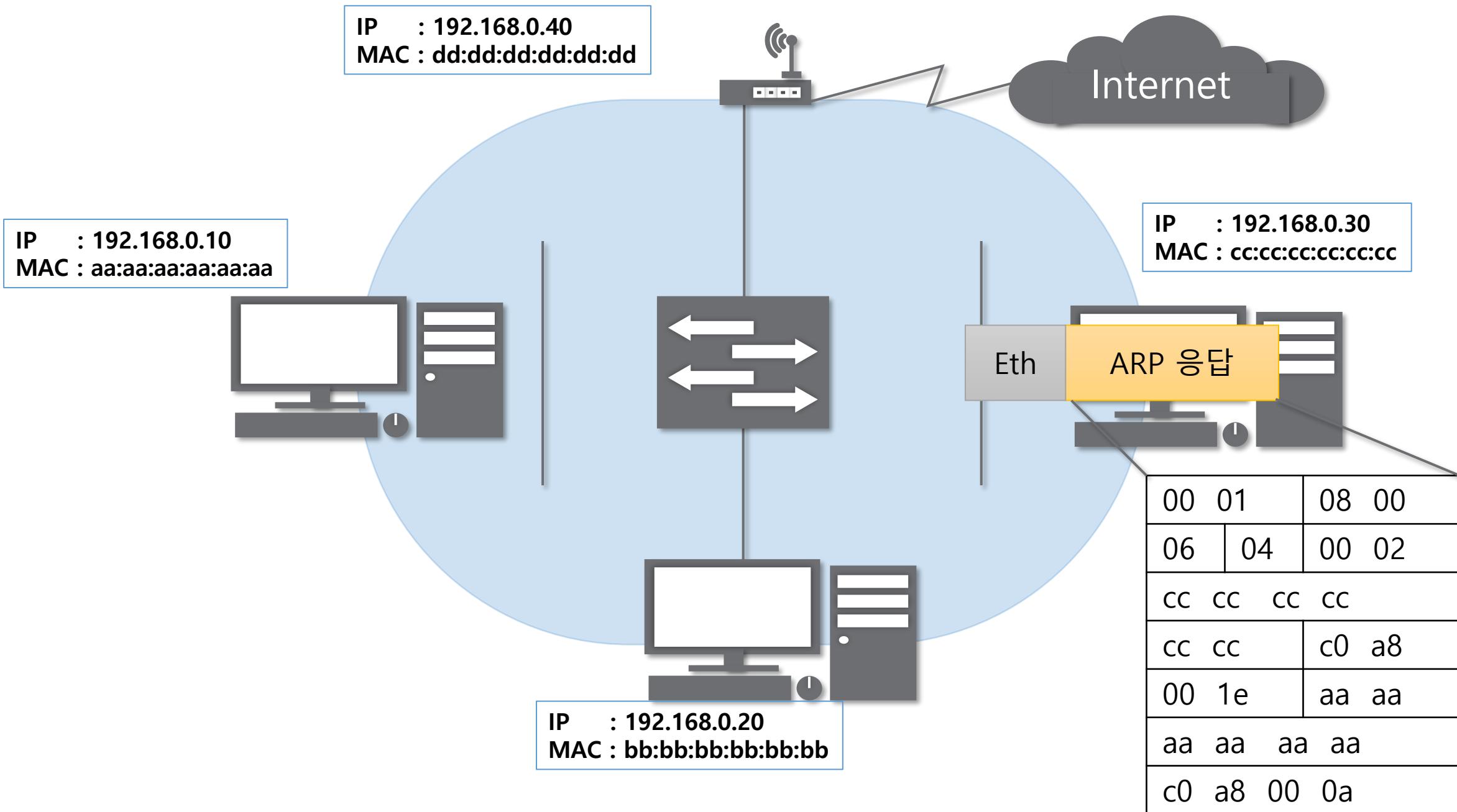


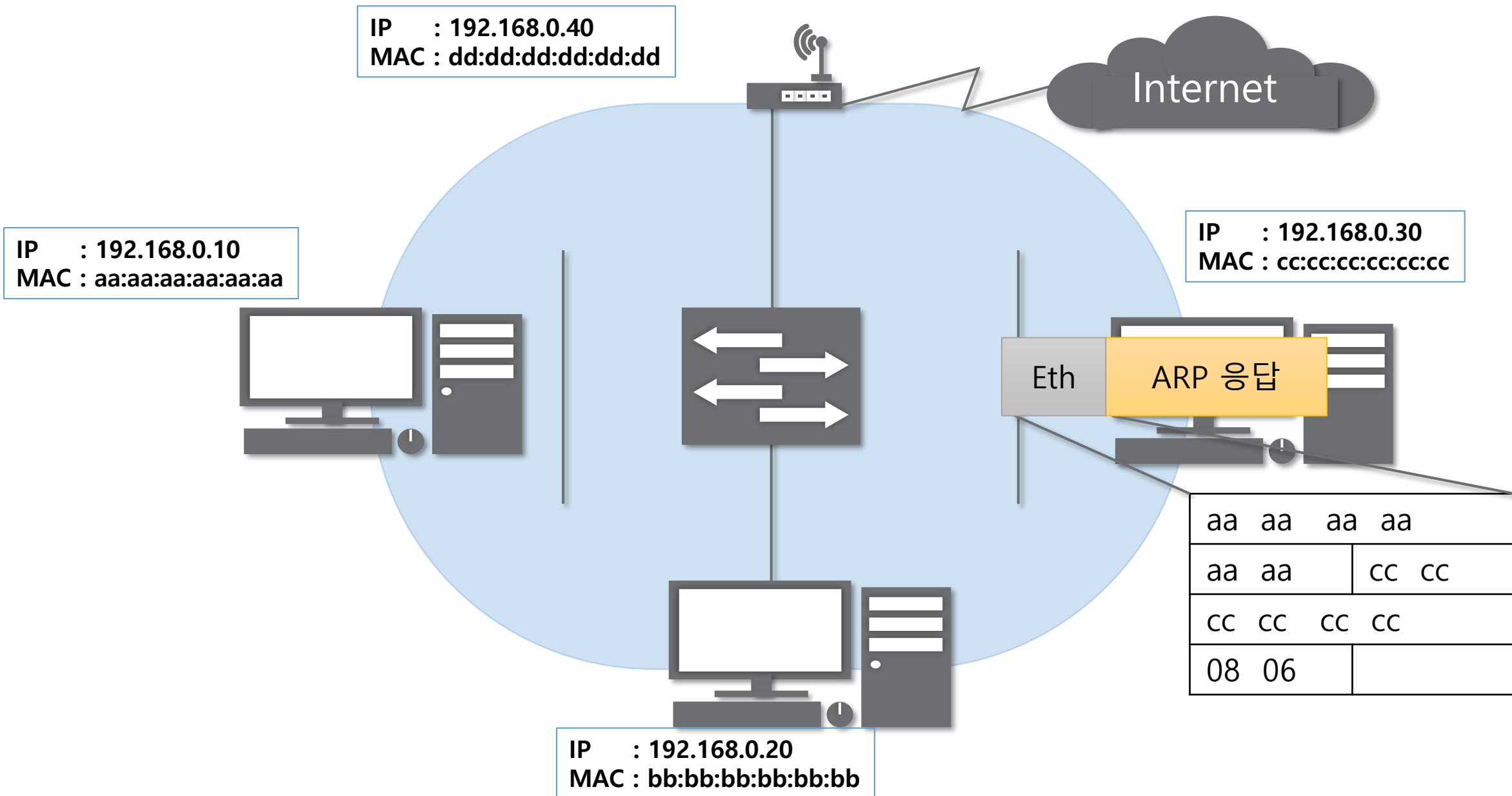
IP : 192.168.0.20  
MAC : bb:bb:bb:bb:bb:bb











IP : 192.168.0.40  
MAC : dd:dd:dd:dd:dd:dd

Internet

## ARP 캐시 테이블

192.168.0.40  
dd:dd:dd:dd:dd:dd

192.168.0.20  
bb:bb:bb:bb:bb:bb

192.168.0.30  
cc:cc:cc:cc:cc:cc

IP : 192.168.0.10  
MAC : aa:aa:aa:aa:aa:aa

IP : 192.168.0.30  
MAC : cc:cc:cc:cc:cc:cc



aa aa aa aa	00 01	08 0e	
aa aa cc cc	c0 06	04 02	
cc cc cc cc	cc cc cc cc		
08 06		cc cc	c0 a8
	00 1e	aa aa	
aa aa aa aa		IP : 192.168.0.20 MAC : bb:bb:bb:bb:bb:bb	
c0 a8 00 0a			

# ARP 테이블

# ARP 테이블

나와 통신했던 컴퓨터들

“

통신했던 컴퓨터들의 주소는  
ARP 테이블  
에 남는다.

“

```
관리자: C:\Windows\system32\cmd.exe
C:\>arp -a

인터페이스: 192.168.0.189 --- 0xc
인터넷 주소          물리적 주소      유형
192.168.0.1           90-9f-33-df-14-e8  동적
192.168.0.4           e8-03-9a-68-98-21  동적
192.168.0.7           e8-11-32-34-7a-9f  동적
192.168.0.8           e8-11-32-33-01-b5  동적
192.168.0.11          e8-11-32-34-d4-58  동적
192.168.0.12          e8-11-32-34-5f-ad  동적
192.168.0.15          e8-11-32-34-60-a9  동적
192.168.0.16          e8-03-9a-65-da-c1  동적
192.168.0.17          e8-11-32-34-61-7e  동적
192.168.0.18          14-c2-13-e7-34-14  동적
```

실습

## 1. ARP 테이블 확인해보기

1. 윈도우에서 간단하게 내PC의 ARP 테이블을 확인해보기

## 2. ARP 프로토콜 분석하기

1. Wireshark를 이용해서 ARP 프로토콜을 캡쳐하고 분석해보기

멀리 있는 컴퓨터끼리는  
이렇게 데이터를 주고받는다

# 목차

## INDEX

### IPv4 프로토콜

IPv4가 하는 일  
IPv4 프로토콜의 구조

### ICMP 프로토콜

ICMP가 하는 일  
ICMP 프로토콜의 구조

### 라우팅 테이블

내가 보낸 패킷은  
어디로 가는가

### 다른 네트워크와 통신 과정

다른 네트워크까지  
내 패킷의 이동 과정

### IPv4의 조각화

조각화란?  
조각화하는 과정

### 실습

라우팅 테이블 확인해보기  
패킷 분석하기

# **IPv4 프로토콜**

# IPv4 프로토콜

## IPv4가 하는 일

네트워크 상에서 데이터를 교환하기 위한 프로토콜

데이터가 **정확하게 전달될 것을 보장하지 않는다.**

중복된 패킷을 전달하거나 패킷의 순서를 잘못 전달할 가능성도 있다.  
(악의적으로 이용되면 DoS 공격이 됨)

데이터의 정확하고 순차적인 전달은 그보다 상위 프로토콜인  
TCP에서 보장한다.

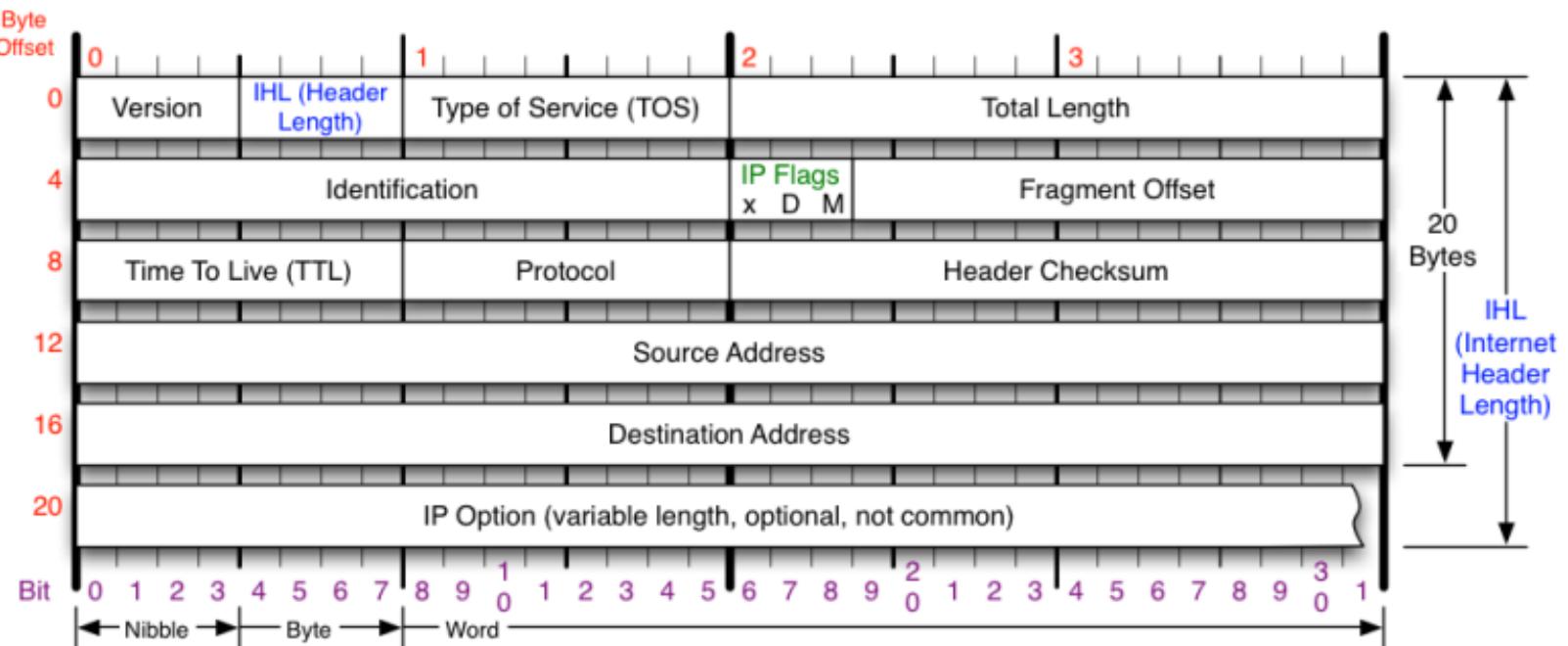
# IPv4 프로토콜

## IPv4 프로토콜의 구조

//

다른 네트워크의 특정 대상을 찾는  
IPv4 프로토콜

//



# **ICMP 프로토콜**

# ICMP 프로토콜

## ICMP가 하는 일

ICMP (Internet Control Message Protocol, 인터넷 제어 메시지 프로토콜)

네트워크 컴퓨터 위에서 돌아가는 운영체제에서 **오류 메시지를** 전송 받는 데 주로 쓰인다.

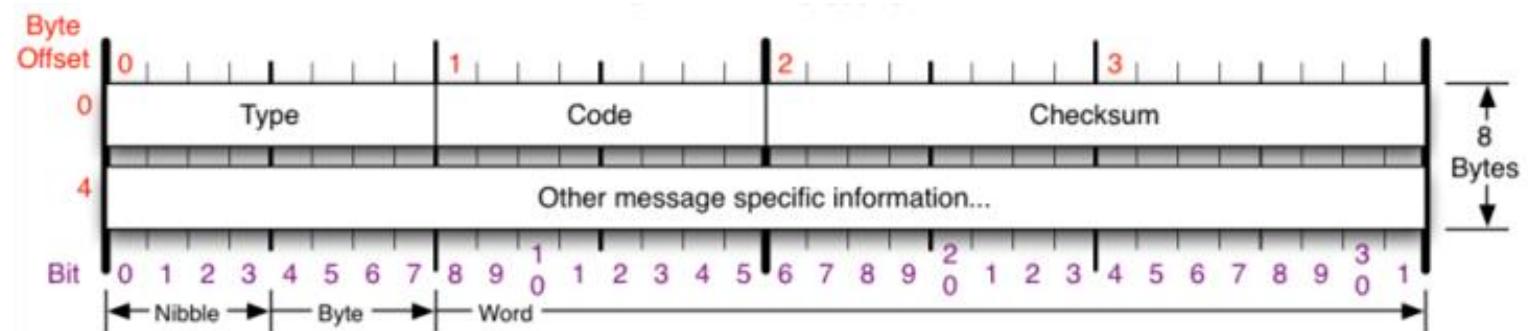
프로토콜 구조의 Type과 Code를 통해 오류 메시지를 전송 받는다.

# ICMP 프로토콜

## ICMP 프로토콜의 구조

//

특정 대상과 내가 통신이 잘되는지 확인하는  
ICMP 프로토콜



//

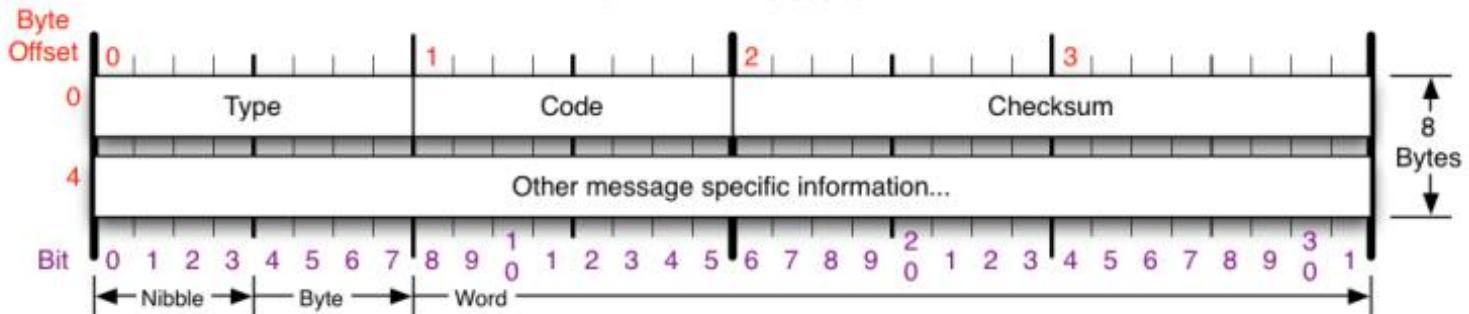
# ICMP 프로토콜

## ICMP 프로토콜의 구조

//

특정 대상과 내가 통신이 잘되는지 확인하는  
ICMP 프로토콜

//



ICMP Message Types		Checksum	
Type	Code/Name	Checksum	
0	Echo Reply	Checksum of ICMP header	
3	Destination Unreachable	RFC 792	
0	Net Unreachable	Please refer to RFC 792 for the Internet Control Message protocol (ICMP) specification.	
1	Host Unreachable		
2	Protocol Unreachable		
3	Port Unreachable		
4	Fragmentation required, and DF set		
5	Source Route Failed		
6	Destination Network Unknown		
7	Destination Host Unknown		
8	Source Host Isolated		
9	Network Administratively Prohibited		
10	Host Administratively Prohibited		
11	Network Unreachable for TOS		
3	Destination Unreachable (continued)	11 Time Exceeded	
12	Host Unreachable for TOS	0 TTL Exceeded	
13	Communication Administratively Prohibited	1 Fragment Reassembly Time Exceeded	
4	Source Quench	12 Parameter Problem	
5	Redirect	0 Pointer Problem	
0	Redirect Datagram for the Network	1 Missing a Required Operand	
1	Redirect Datagram for the Host	2 Bad Length	
2	Redirect Datagram for the TOS & Network	13 Timestamp	
3	Redirect Datagram for the TOS & Host	14 Timestamp Reply	
8	Echo	15 Information Request	
9	Router Advertisement	16 Information Reply	
10	Router Selection	17 Address Mask Request	
18	Address Mask Reply	18 Address Mask Reply	
30	Traceroute	30 Traceroute	

**라우팅 테이블**

# 라우팅 테이블

내가 보낸 패킷은 어디로 가는가

어디로 보내야 하는지 설정되어 있는  
라우팅 테이블

〃

〃

```
관리자: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>netstat -r
=====
IPv4 경로 테이블
=====
활성 경로:
네트워크 대상 네트워크 마스크     게이트웨이     인터페이스     메트릭
          0.0.0.0   0.0.0.0      192.168.0.1  192.168.0.189    276
          127.0.0.0  255.0.0.0     연결됨        127.0.0.1     306
          127.0.0.1  255.255.255.255  연결됨        127.0.0.1     306
          127.255.255.255  255.255.255.255  연결됨        127.0.0.1     306
          192.168.0.0  255.255.255.0     연결됨        192.168.0.189    276
          192.168.0.189  255.255.255.255  연결됨        192.168.0.189    276
          192.168.0.255  255.255.255.255  연결됨        192.168.0.189    276
          192.168.1.0   255.255.255.0     연결됨        192.168.0.189    276
```

# **다른 네트워크와 통신 과정**

# 다른 네트워크와 통신 과정

다른 네트워크까지 내 패킷의 이동 과정

---

“

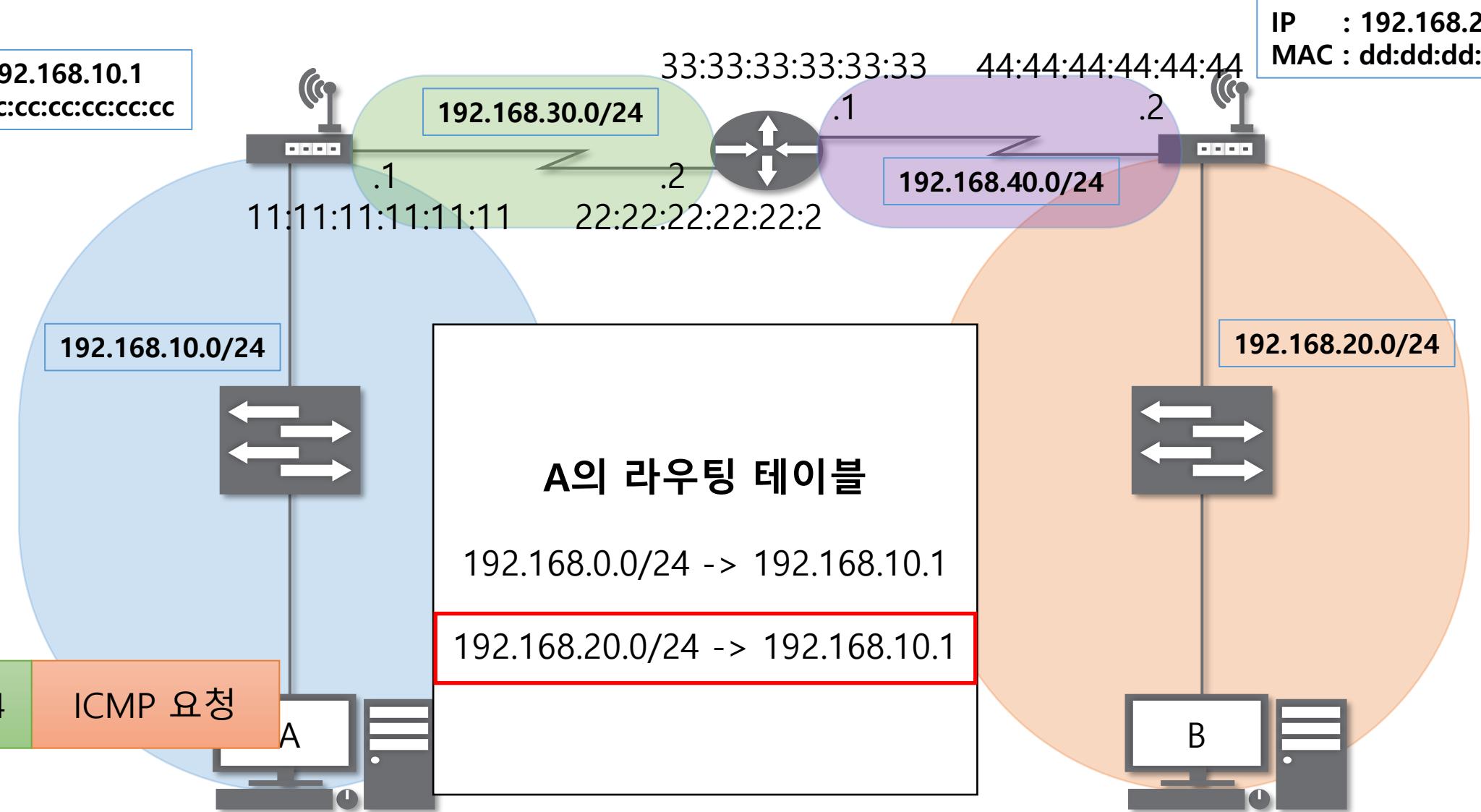
내 컴퓨터에서 보낸 패킷이  
다른 네트워크의 컴퓨터까지  
어떻게 이동하는가

”

---

IP : 192.168.10.1  
MAC : cc:cc:cc:cc:cc:cc

IP : 192.168.20.1  
MAC : dd:dd:dd:dd:dd:dd

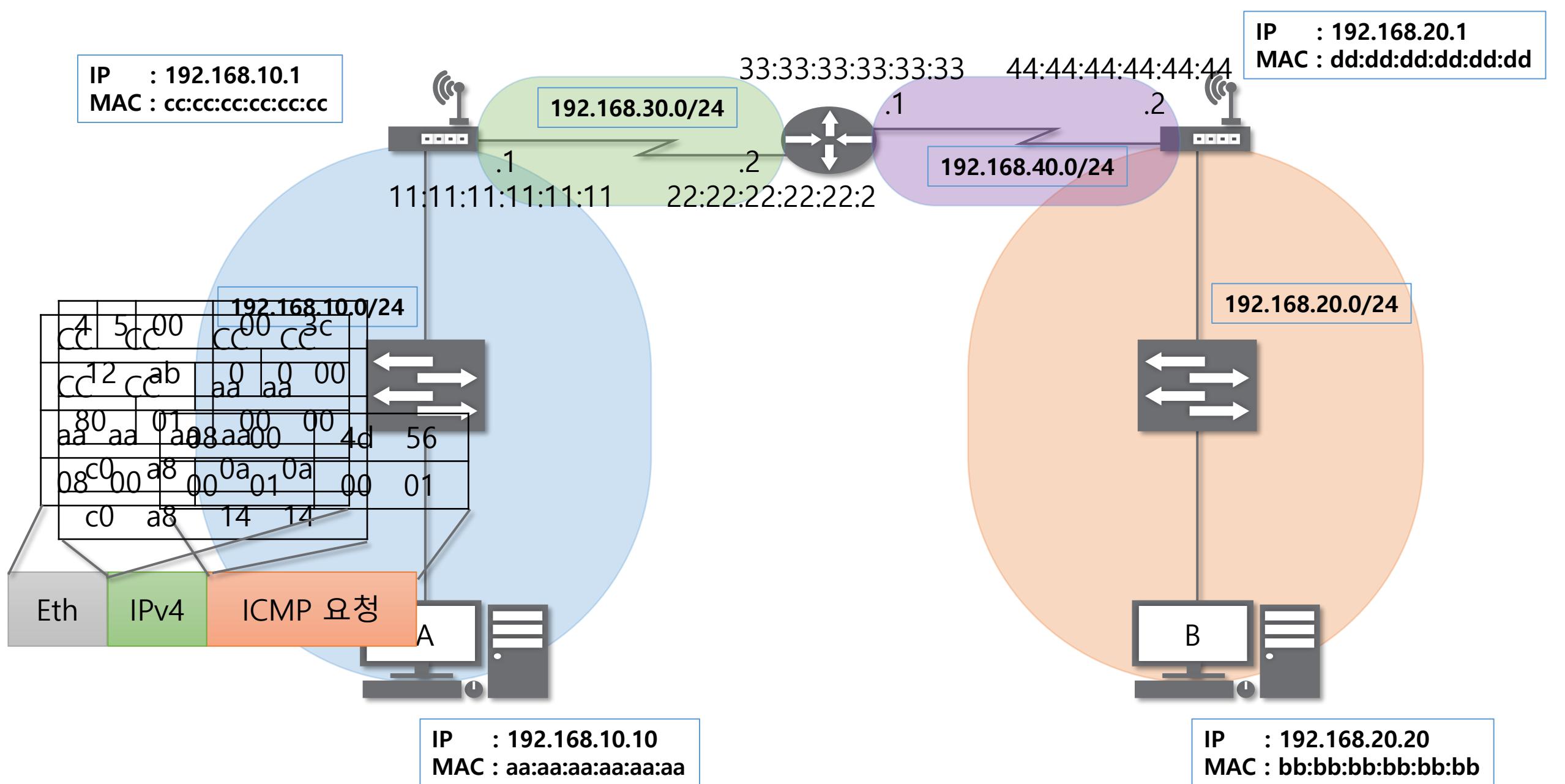


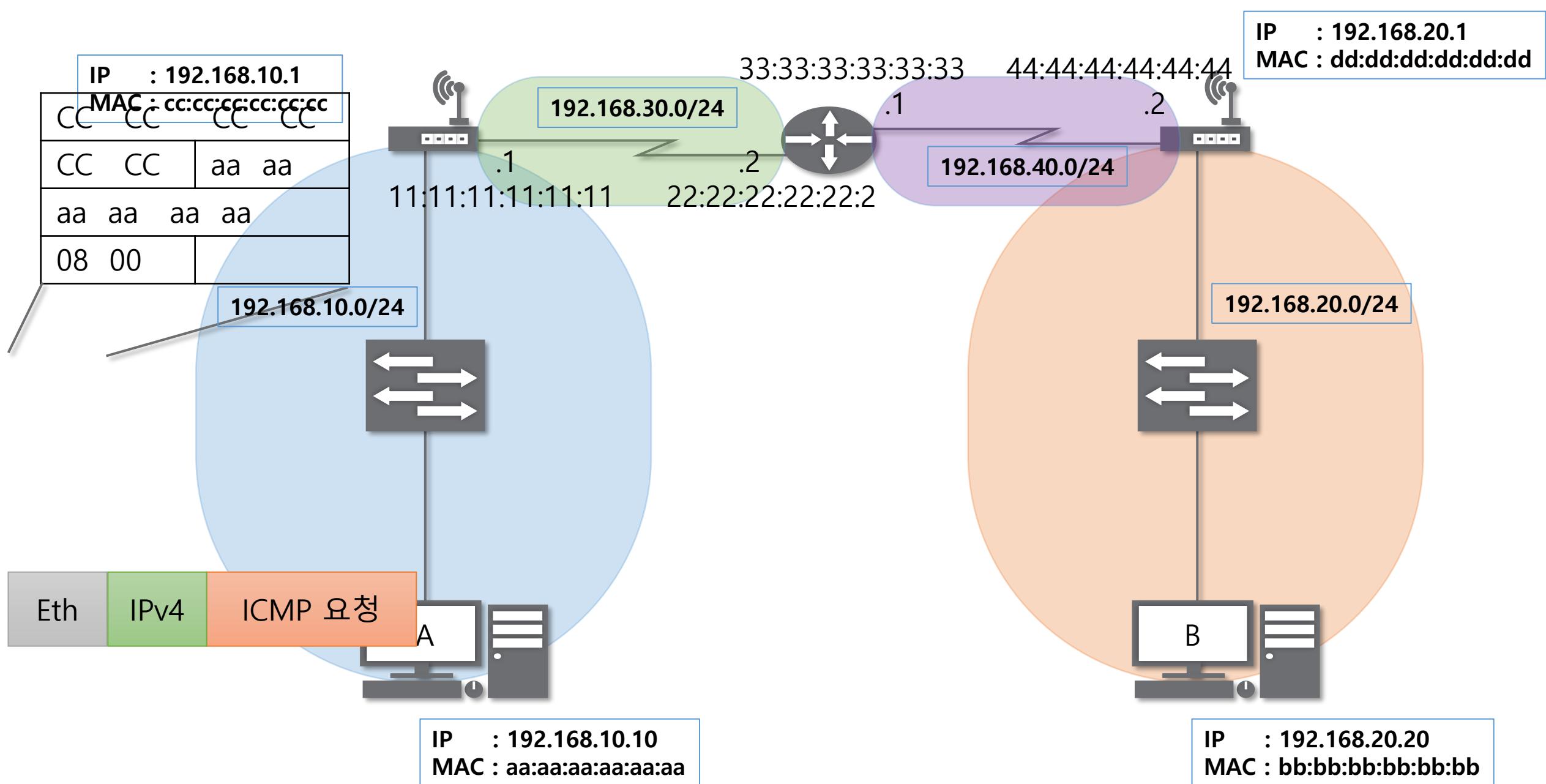
IP : 192.168.10.10  
MAC : aa:aa:aa:aa:aa:aa

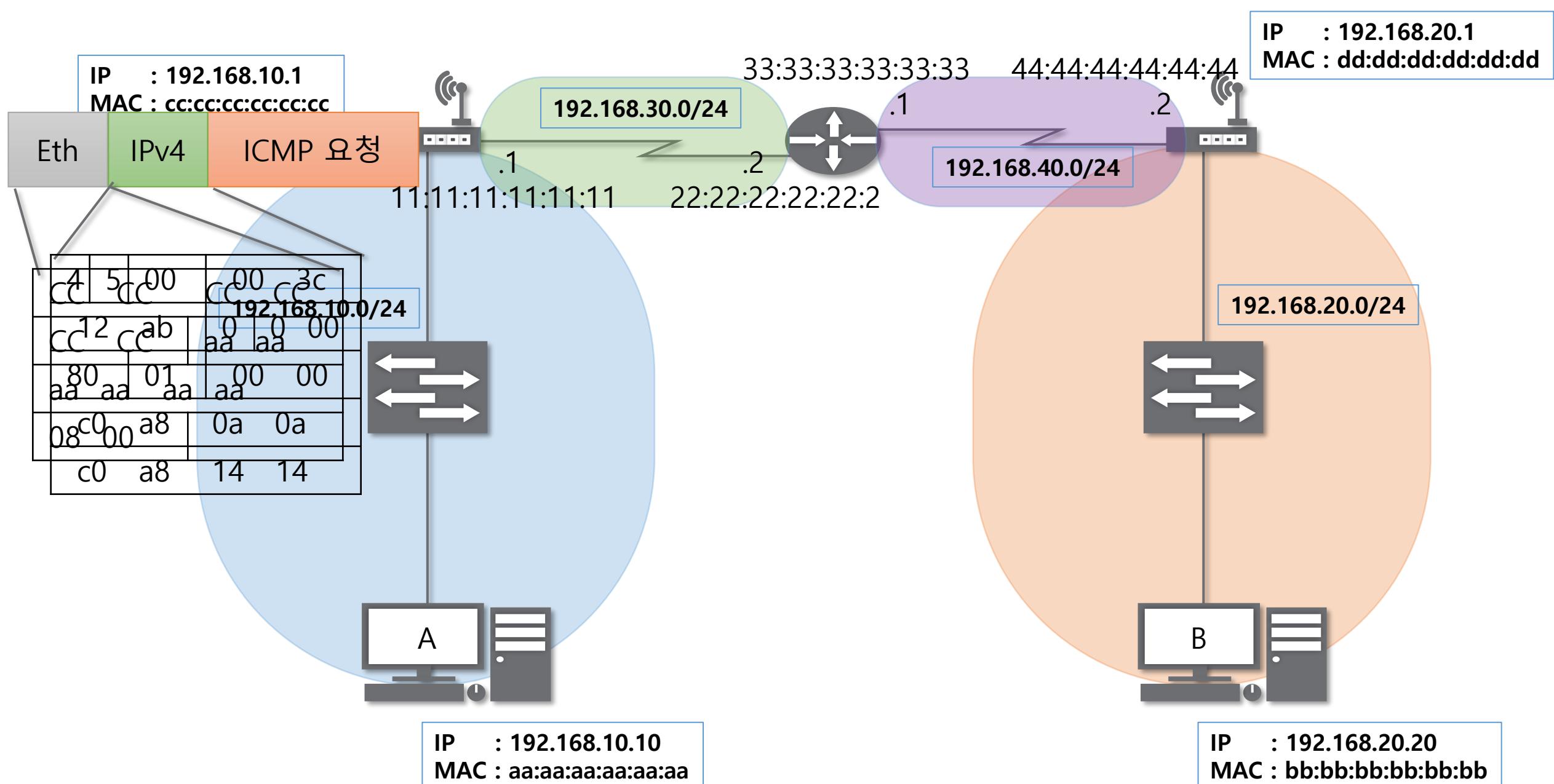
IP : 192.168.20.20  
MAC : bb:bb:bb:bb:bb:bb

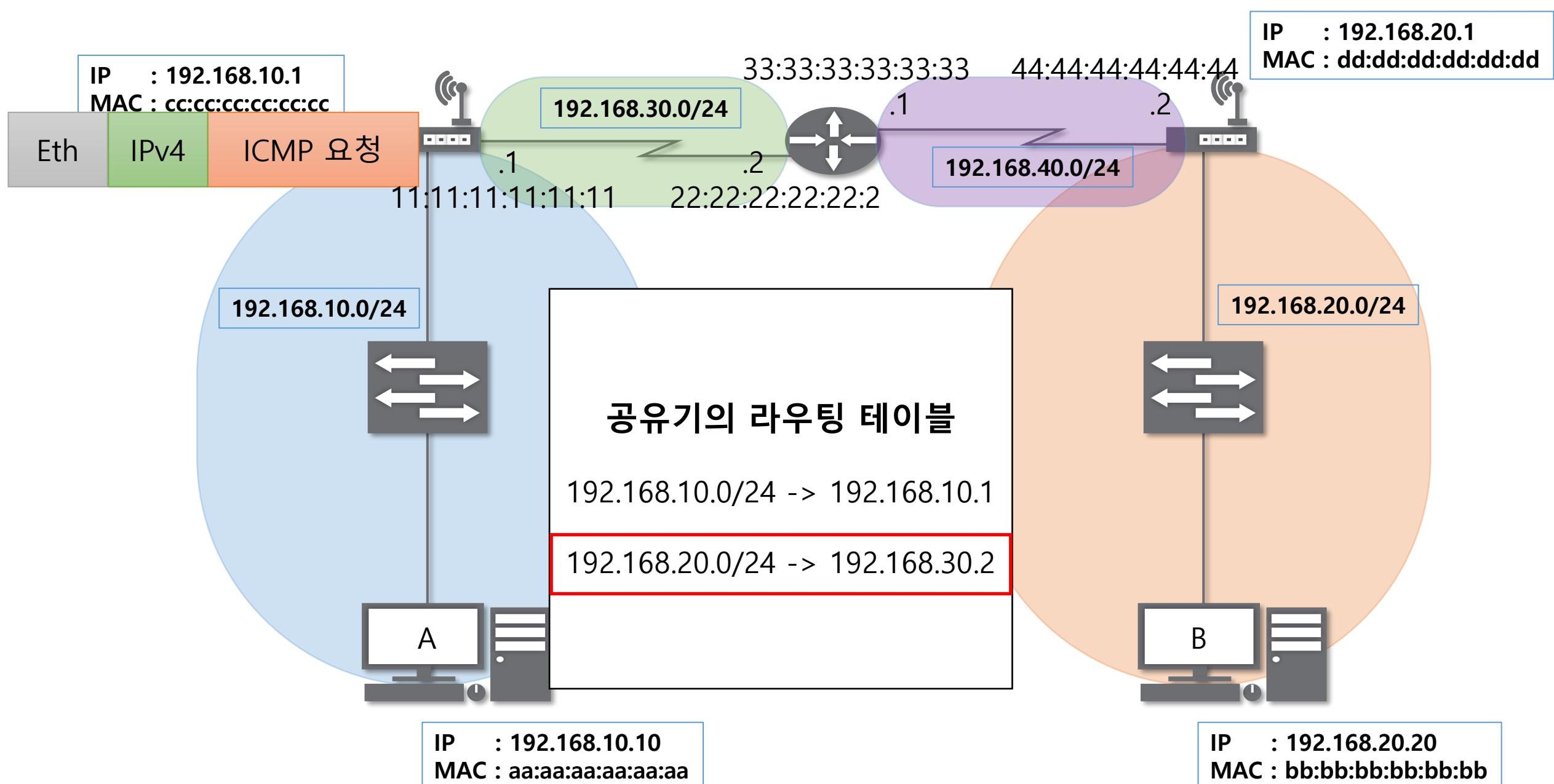
IP : 192.168.10.1  
MAC : cc:cc:cc:cc:cc:cc

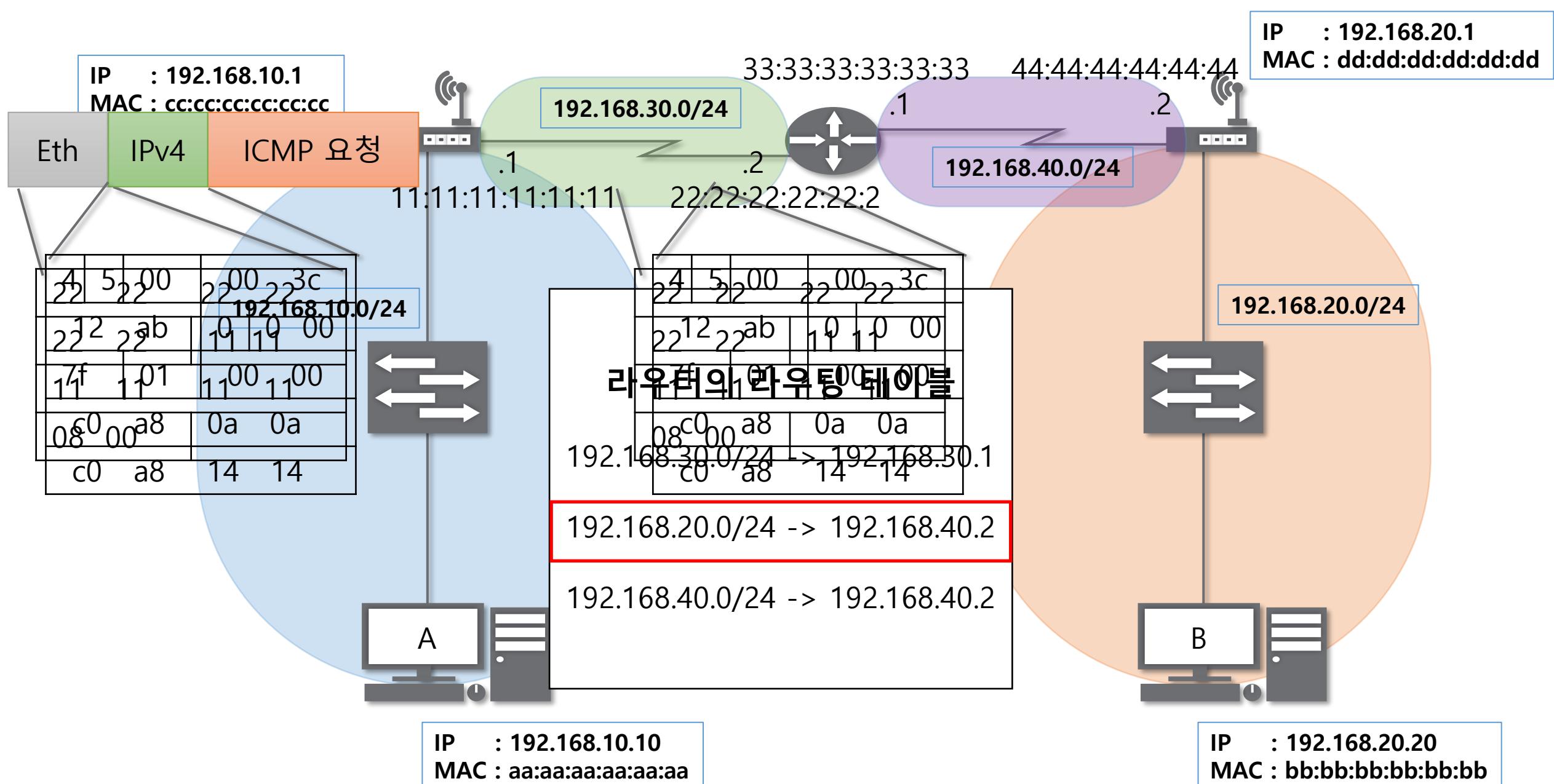
IP : 192.168.20.1  
MAC : dd:dd:dd:dd:dd:dd



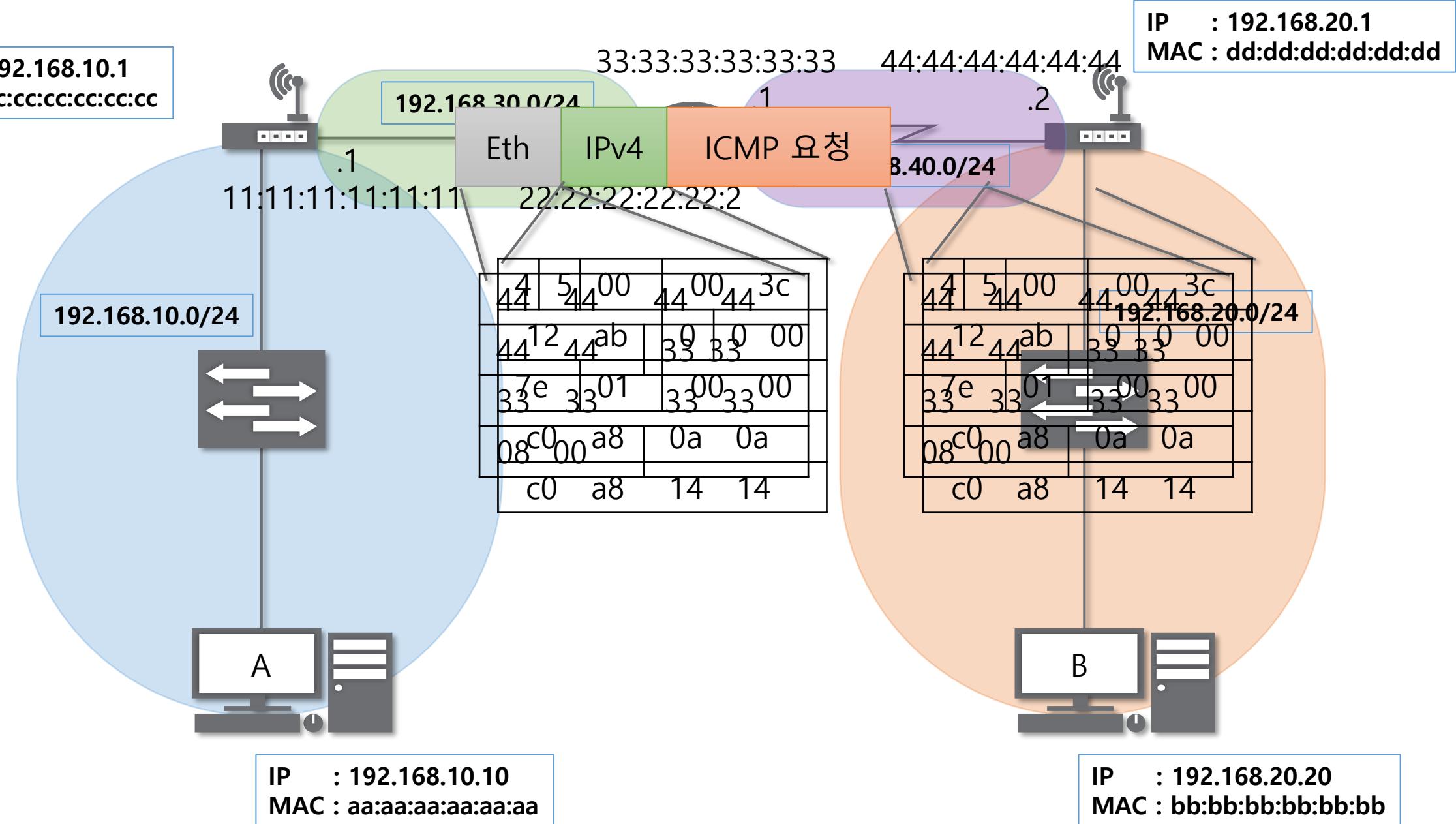




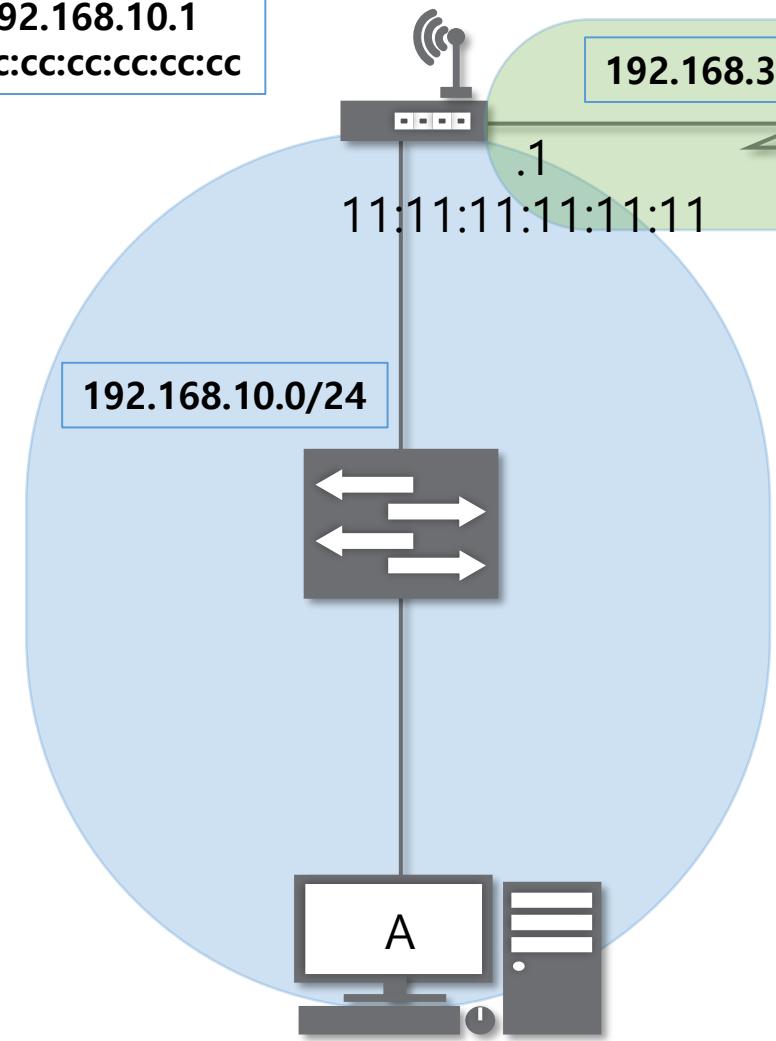




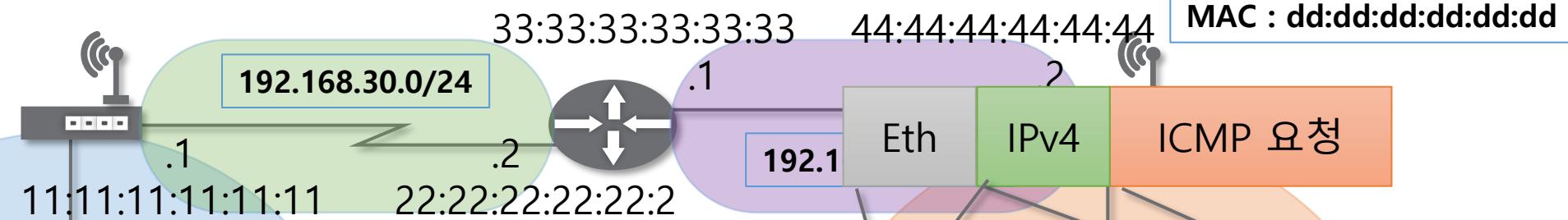
IP : 192.168.10.1  
MAC : cc:cc:cc:cc:cc:cc



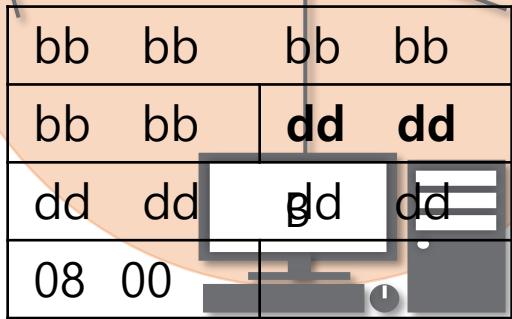
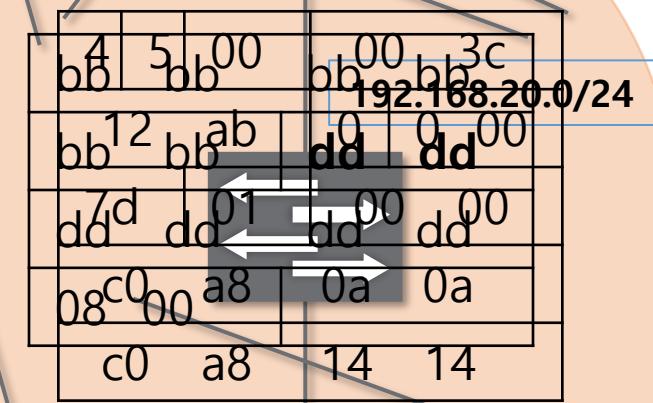
IP : 192.168.10.1  
MAC : cc:cc:cc:cc:cc:cc



IP : 192.168.10.10  
MAC : aa:aa:aa:aa:aa:aa

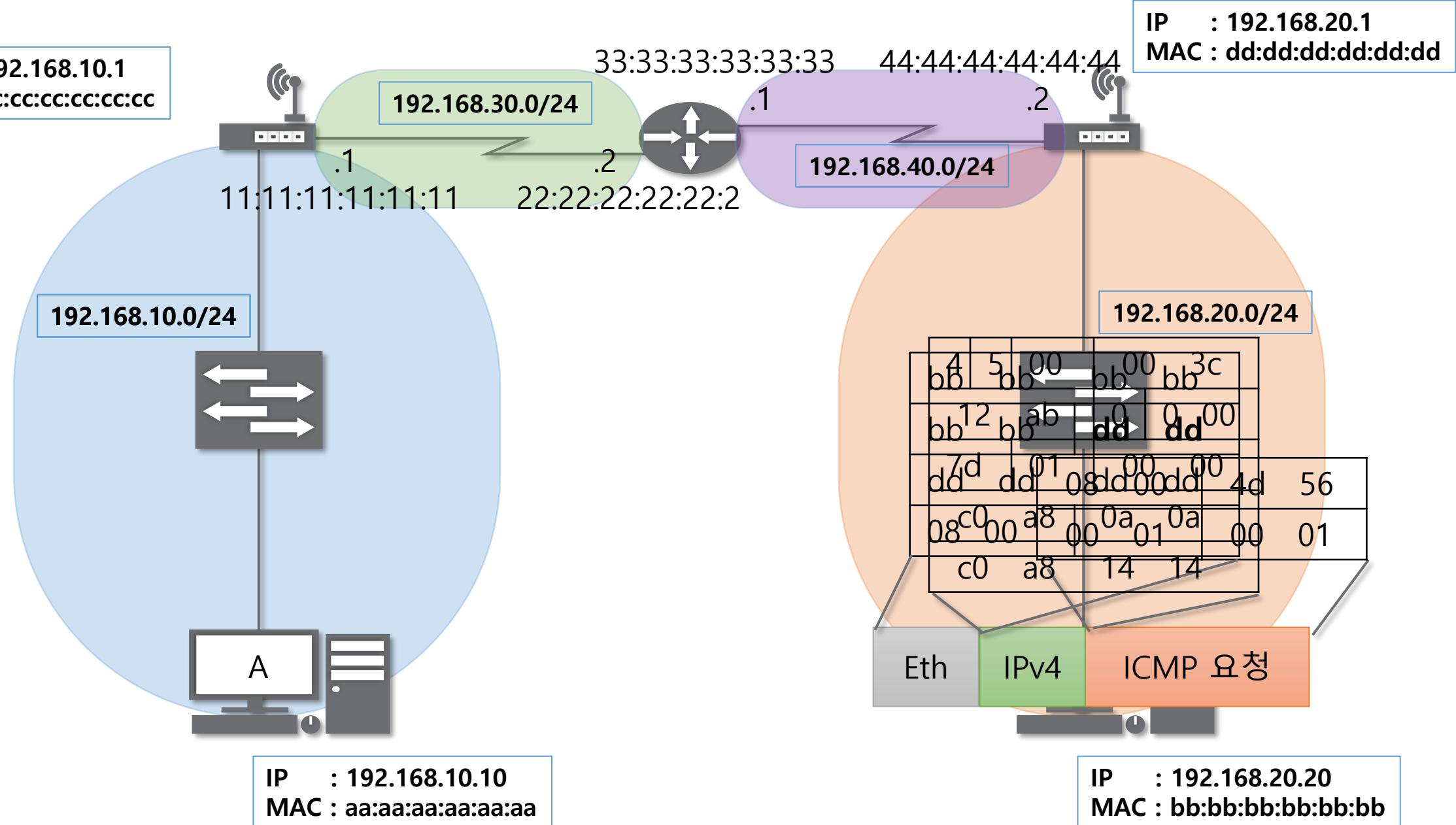


IP : 192.168.20.1  
MAC : dd:dd:dd:dd:dd:dd

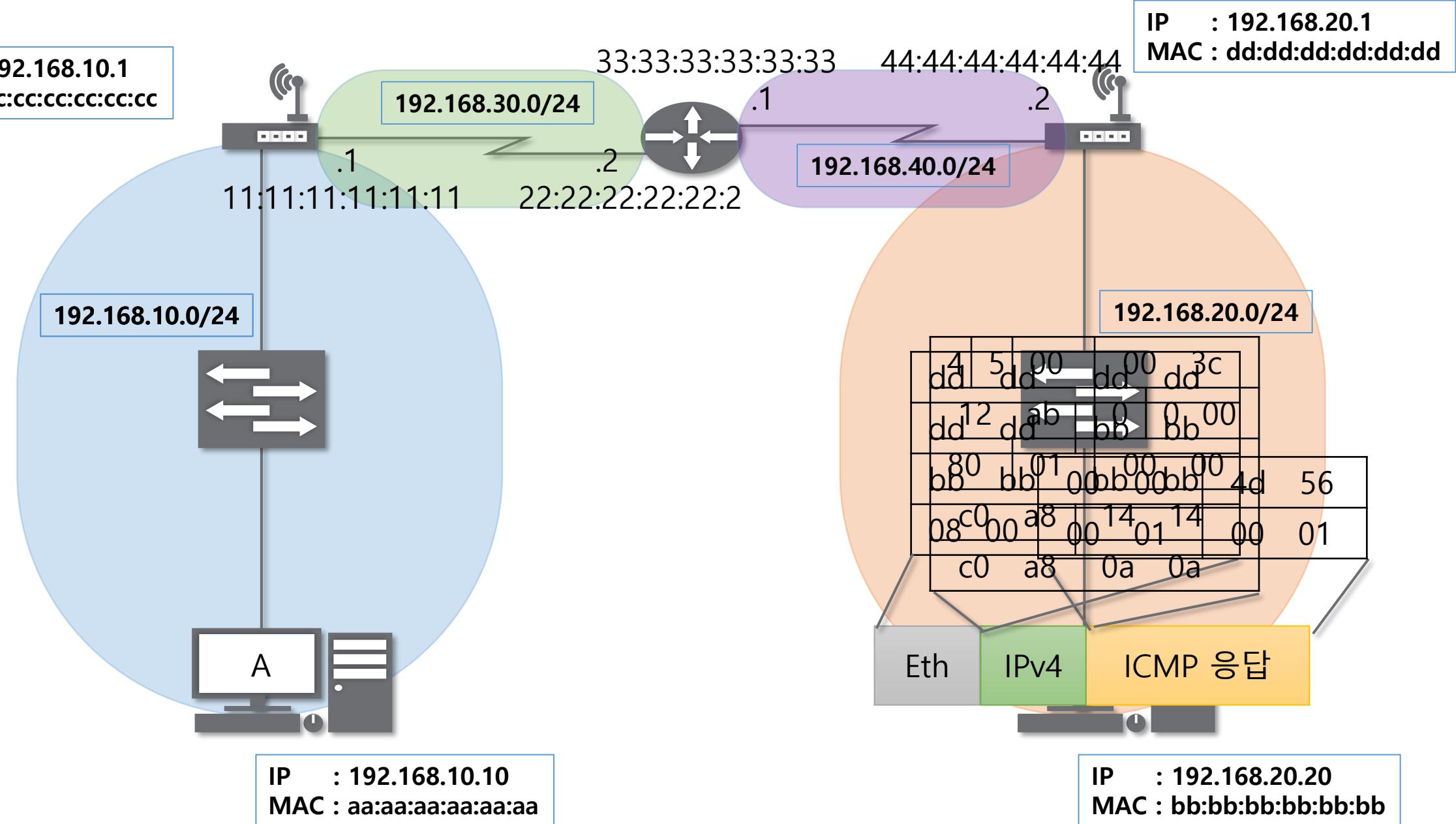


IP : 192.168.20.20  
MAC : bb:bb:bb:bb:bb:bb

IP : 192.168.10.1  
MAC : cc:cc:cc:cc:cc:cc



IP : 192.168.10.1  
MAC : cc:cc:cc:cc:cc:cc



# **IPv4의 조각화**

# IPv4의 조각화

## 조각화란?

큰 IP 패킷들이 적은 MTU(Maximum Transmission Unit)를 갖는 링크를 통하여 전송되려면 여러 개의 작은 패킷으로 쪼개어/조각화 되어 전송돼야 한다.

즉, 목적지까지 패킷을 전달하는 과정에 통과하는 각 라우터마다 전송에 적합한 프레임으로 변환이 필요하다.

일단 조각화되면, 최종 목적지에 도달할 때까지 재조립되지 않는 것이 일반적이다.

IPv4에서는 발신지 뿐만 아니라 중간 라우터에서도 IP 조각화가 가능

IPv6에서는 IP 단편화가 발신지에서 만 가능

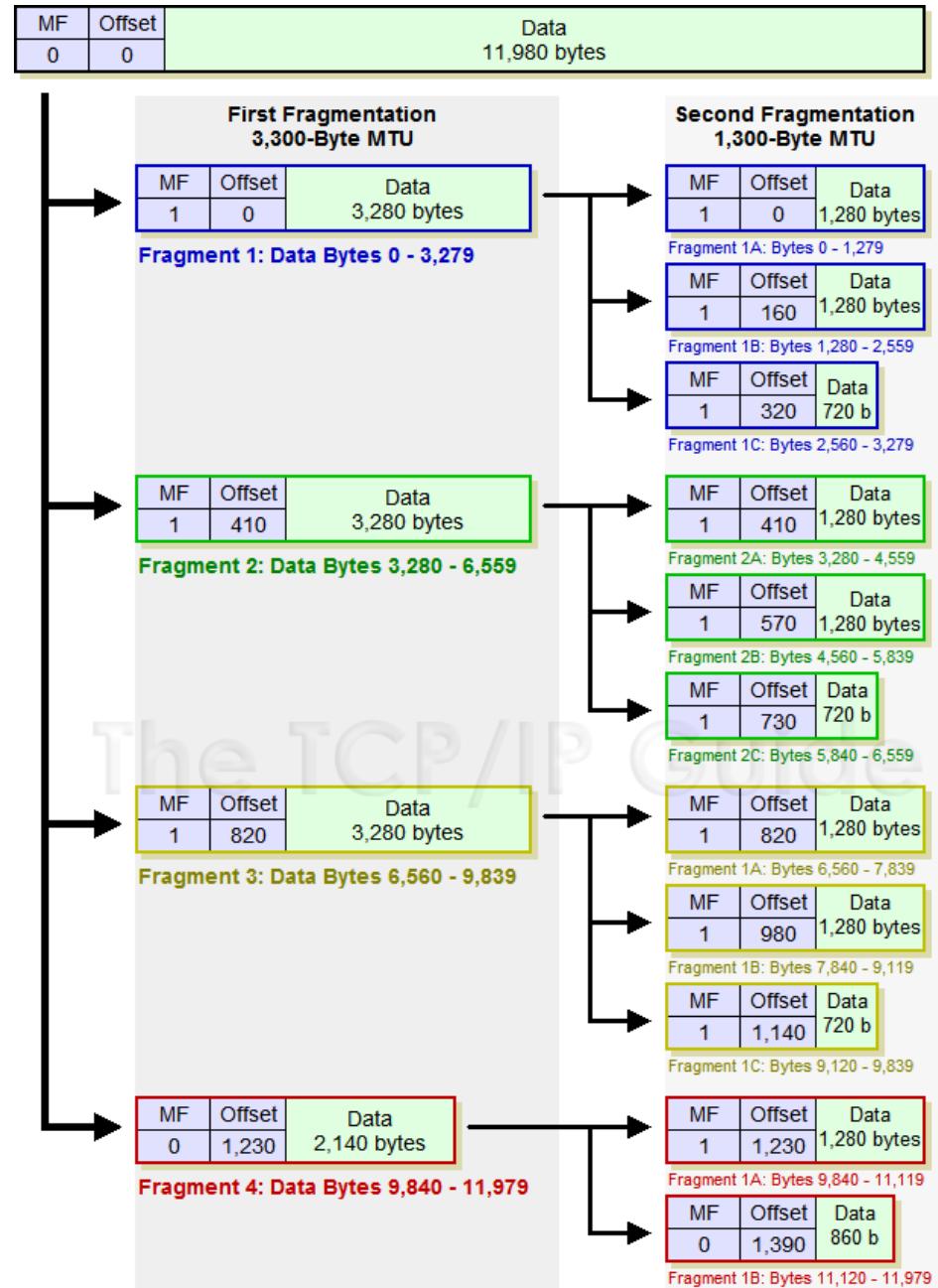
재조립은 항상 최종 수신지에서 만 가능함

# IPv4의 조각화란? 조각화란?

//

여러 개의 패킷으로  
조각화 된 패킷

//



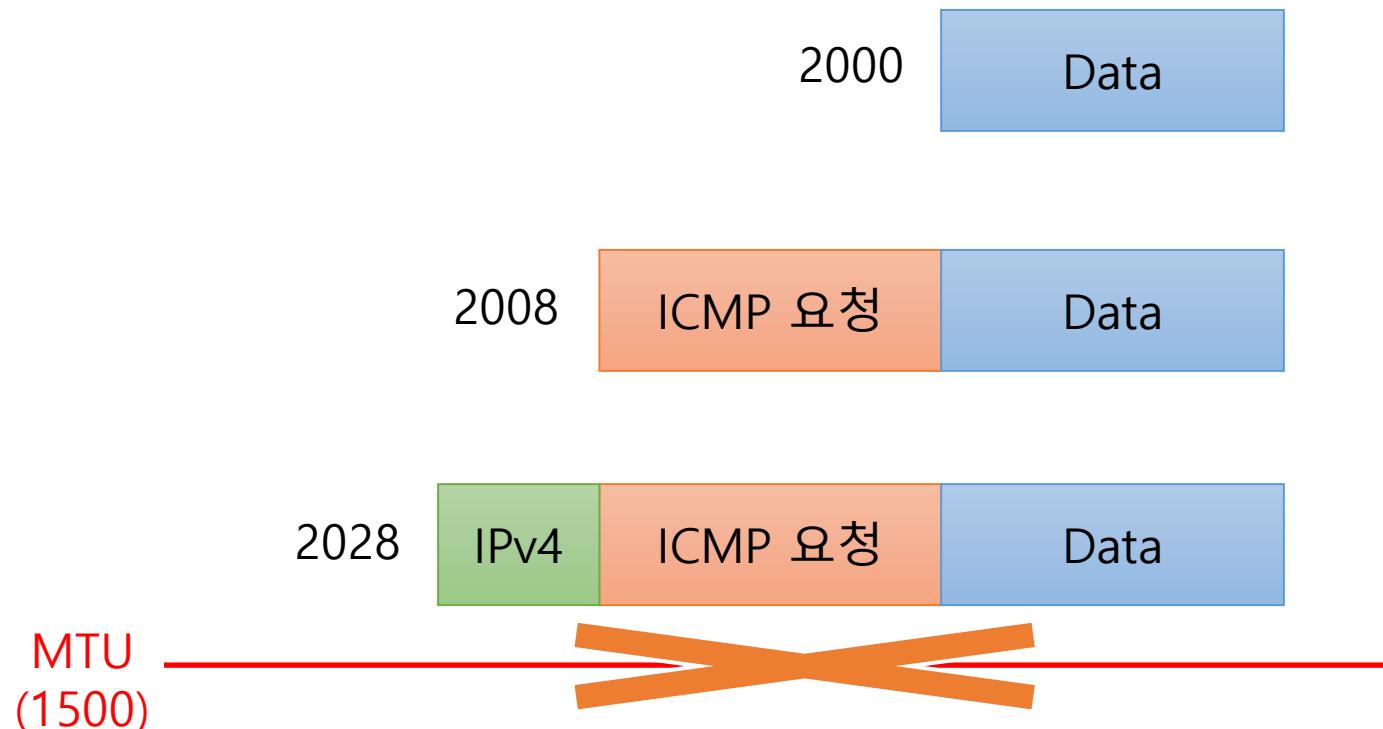
# IPv4의 조각화

큰 데이터를 전송하는 패킷이 조각화하는 과정

〃

큰 데이터를 보낼 때 패킷이  
조각화하는 과정

〃



# IPv4의 조각화

큰 데이터를 전송하는 패킷이 조각화하는 과정

2000

Data

“

큰 데이터를 보낼 때 패킷이  
조각화하는 과정

”

MTU  
(1500)

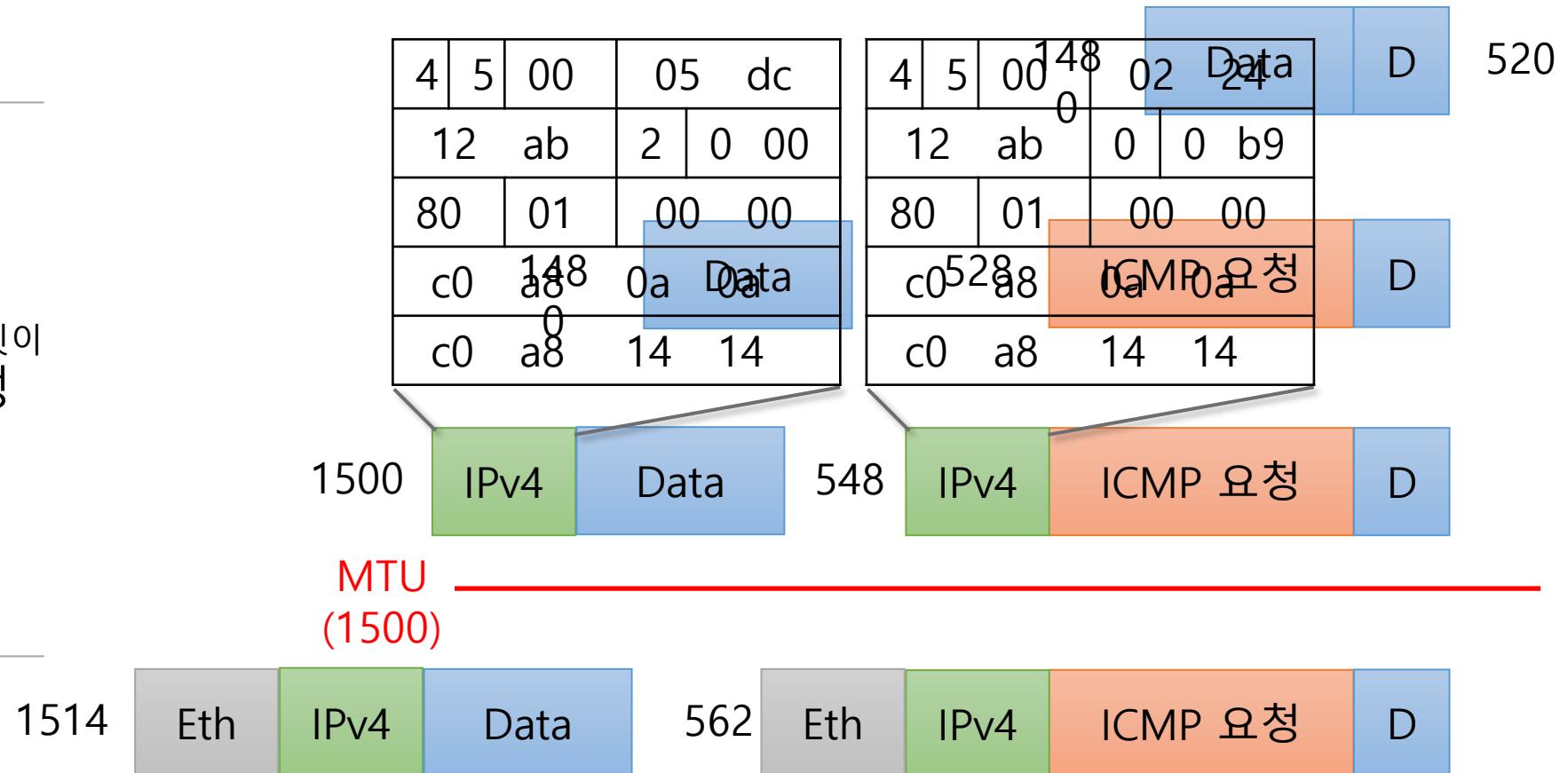
# IPv4의 조각화

큰 데이터를 전송하는 패킷이 조각화하는 과정

〃

큰 데이터를 보낼 때 패킷이  
조각화하는 과정

〃



실습

## 1. 라우팅 테이블 확인해보기

1. 윈도우에서 간단하게 내PC의 라우팅 테이블을 확인해보기

## 2. 패킷 분석하기

1. Wireshark를 이용해서 다른 네트워크 대역으로 보낸 내 패킷
  2. 캡쳐하고 분석해보기

컴퓨터의 프로그램끼리는  
이렇게 데이터를 주고 받는다

# 목차

## INDEX

### 4계층 프로토콜

4계층에서 하는 일  
4계층 프로토콜의 종류

### 포트 번호

포트번호의 특징  
Well-Known 포트  
Registered 포트  
Dynamic 포트

### 프로그램의 연결 정보

어떤 프로세스와  
어떤 프로세스가  
연결되어 있는지 확인

### 실습

현재 연결 상태 확인하기  
특정 서비스의 포트번호 확인하기

# 4계층 프로토콜

# 4계층 프로토콜

## 4계층에서 하는 일

전송 계층(Transport layer)은 송신자의 **프로세스**와 수신자의 **프로세스를 연결하는 통신 서비스**를 제공한다.

전송 계층은 연결 지향 데이터 스트림 지원, 신뢰성, 흐름 제어, 그리고 다중화와 같은 편리한 서비스를 제공한다.

전송 프로토콜 중 가장 잘 알려진 것은 연결 지향 전송 방식을 사용하는 전송 제어 프로토콜 (TCP)이다. 보다 단순한 전송에 사용되는 사용자 데이터 그램 프로토콜 (UDP)도 있다.

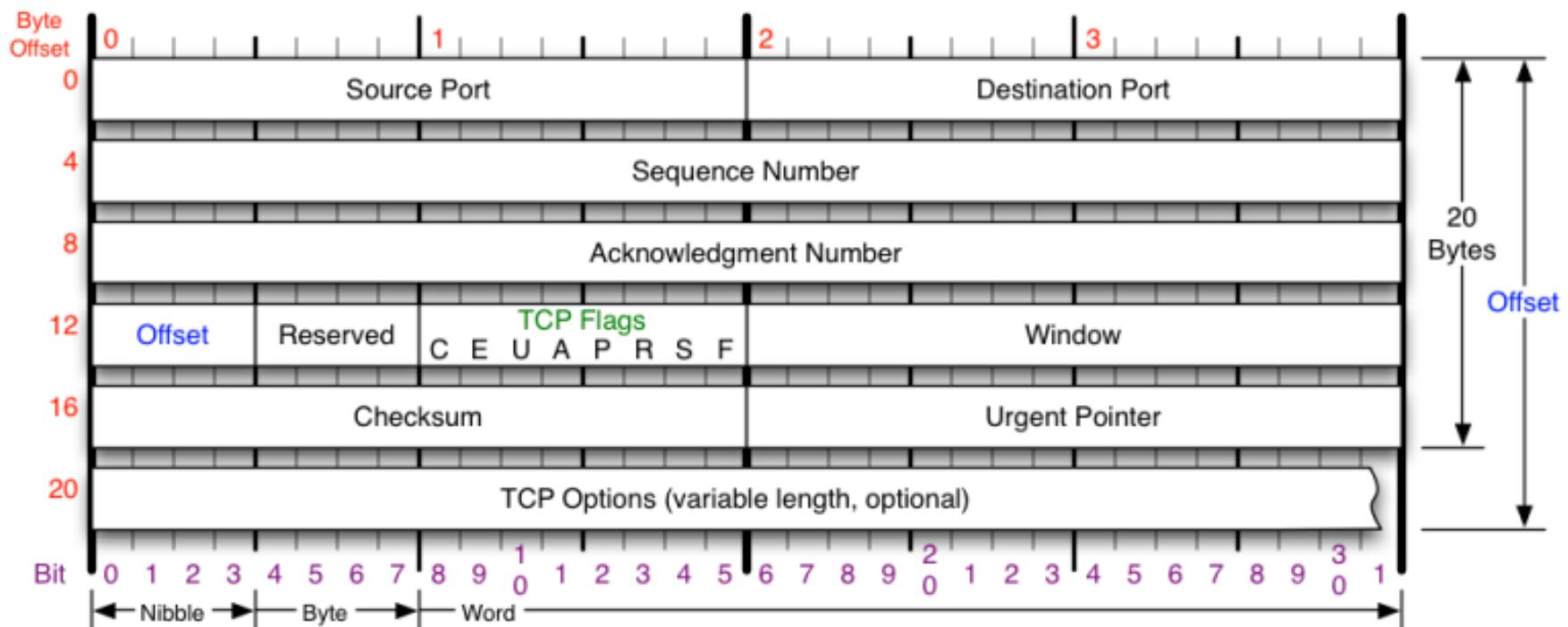
# 4계층 프로토콜

4계층 프로토콜의 종류

//

안전한 연결을 지향하는  
TCP 프로토콜

//

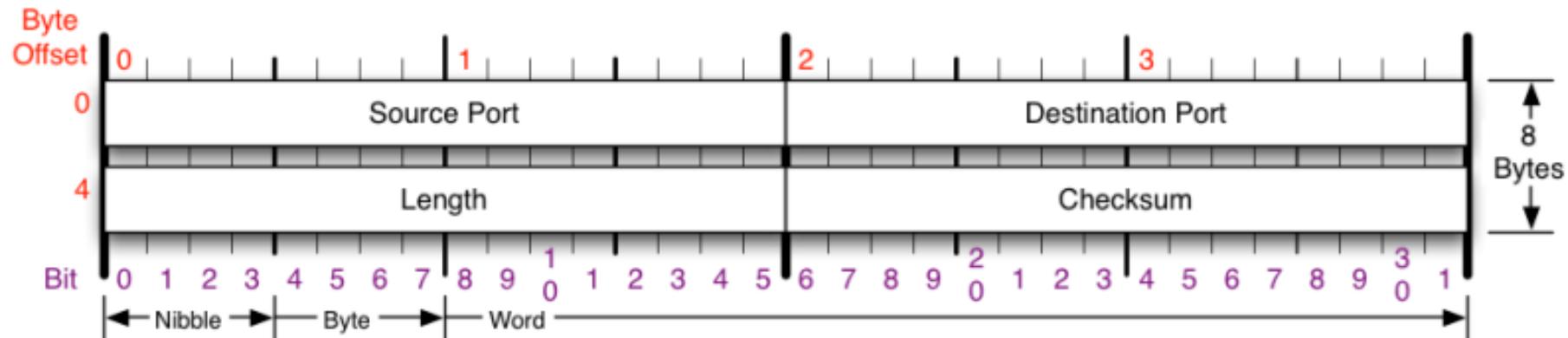


# 4계층 프로토콜

## 4계층 프로토콜의 종류

11

# 안전한 연결을 지향하지 않는 UDP 프로토콜



11

**포트 번호**

# 포트 번호

## 포트 번호의 특징

특정 프로세스와 특정 프로세스가 통신을 하기 위해 사용한다.

하나의 포트는 하나의 프로세스만 사용 가능하다.

하나의 프로세스가 여러 개의 포트를 사용하는 것은 가능하다.

포트 번호는 일반적으로 정해져 있지만 무조건 지켜야 하는 것은 아니다.

예를 들어 일반적으로 웹 서비스는 80번 포트를 사용하지만 웹 서비스가 항상 80번 포트를 사용해야만 하는 것은 아니다.

# 포트 번호

## Well-Known 포트

“

전 세계적으로 유명한  
Well-Known 포트

”

서비스 이름	포트 번호
FTP	20번, 21번
SSH	22번
TELNET	23번
DNS	53번
DHCP	67번, 68번
TFTP	69번
HTTP	80번
HTTPS	443번

# 포트 번호

Registered 포트

---

〃

조금은 유명한  
Registered 포트

서비스 이름	포트 번호
오라클 DB 서버	1521번
MySQL 서버	3306번
MS 원격 데스크톱	3389번

〃

---

# 포트 번호

Dynamic 포트

---

//

일반 사용자들이 사용하는  
Dynamic 포트

시작 포트 번호	마지막 포트 번호
49152번	65535번

//

---

# **프로그램의 연결 정보**

# 프로그램의 연결 정보

나와 현재 연결되어 있는 컴퓨터들

현재 포트 활성 여부를 나타내는  
활성 연결 테이블

〃

〃

```
관리자: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>netstat -ano

활성 연결

  프로토콜  로컬 주소          외부 주소      상태        PID
  TCP        0.0.0.0:22          0.0.0.0:0      LISTENING   1128
  TCP        0.0.0.0:80          0.0.0.0:0      LISTENING   4376
  TCP        0.0.0.0:135         0.0.0.0:0      LISTENING   860
  TCP        0.0.0.0:443         0.0.0.0:0      LISTENING   2864
  TCP        0.0.0.0:445         0.0.0.0:0      LISTENING   4
  TCP        0.0.0.0:554         0.0.0.0:0      LISTENING   4524
  TCP        0.0.0.0:902         0.0.0.0:0      LISTENING   2120
  TCP        0.0.0.0:912         0.0.0.0:0      LISTENING   2120
  TCP        0.0.0.0:2869        0.0.0.0:0      LISTENING   4
  TCP        0.0.0.0:5357        0.0.0.0:0      LISTENING   4
  TCP        0.0.0.0:10240        0.0.0.0:0      LISTENING   4
```

실습

## 1. 현재 연결 상태 확인하기

netstat –ano 명령어를 이용하여 내 컴퓨터와 현재 연결된 다른 컴퓨터들을 확인해보기

## 2. 특정 서비스의 포트번호 확인하기

웹 서비스와 같은 유명한 특정 서비스의 포트번호 확인하기

# Berryz web share 설치

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=celine2011&logNo=220294634787>

➤ Netstat –ano

# **비연결지향형 UDP 프로토콜**

# 목차

## INDEX

### UDP 프로토콜

UDP가 하는 일  
UDP 프로토콜의 구조

### UDP 프로토콜을 사용하는 프로그램

UDP 프로토콜을 사용하는  
대표적인 프로그램들

### 따라 學IT

tftpd 를 사용하여  
데이터 공유해보기

# **UDP 프로토콜**

# UDP 프로토콜

## UDP가 하는 일

사용자 데이터그램 프로토콜(User Datagram Protocol, UDP)은 유니버설 데이터그램 프로토콜(Universal Datagram Protocol)이라고 일컫기도 한다.

UDP의 전송 방식은 너무 단순해서 서비스의 신뢰성이 낮고, 데이터그램 도착 순서가 바뀌거나, 중복되거나, 심지어는 통보 없이 누락시키기도 한다.

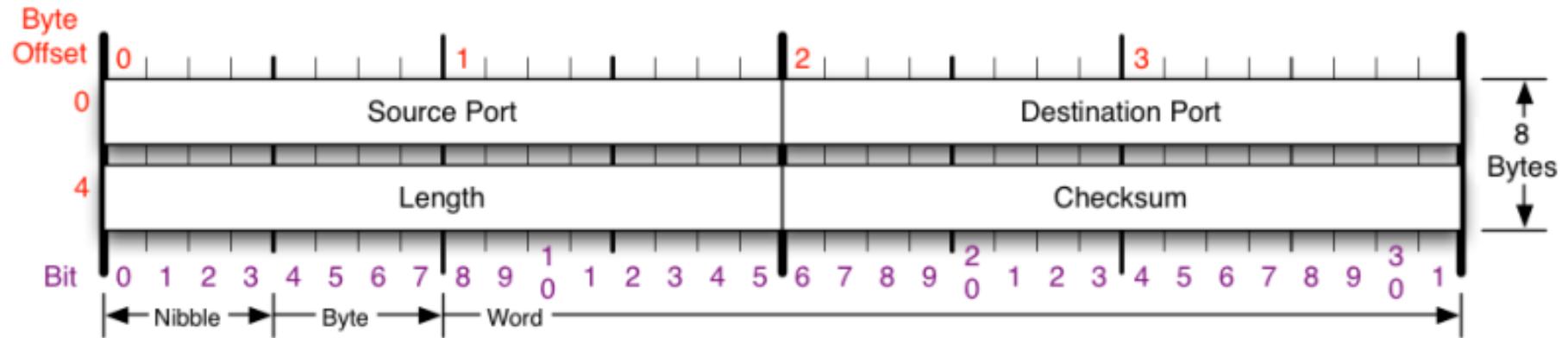
UDP는 일반적으로 오류의 검사와 수정이 필요 없는 프로그램에서 수행할 것으로 가정한다.

# UDP 프로토콜

## UDP 프로토콜의 구조

//

안전한 연결을 지향하지 않는  
UDP 프로토콜



//

# **UDP 프로토콜을 사용하는 프로그램**

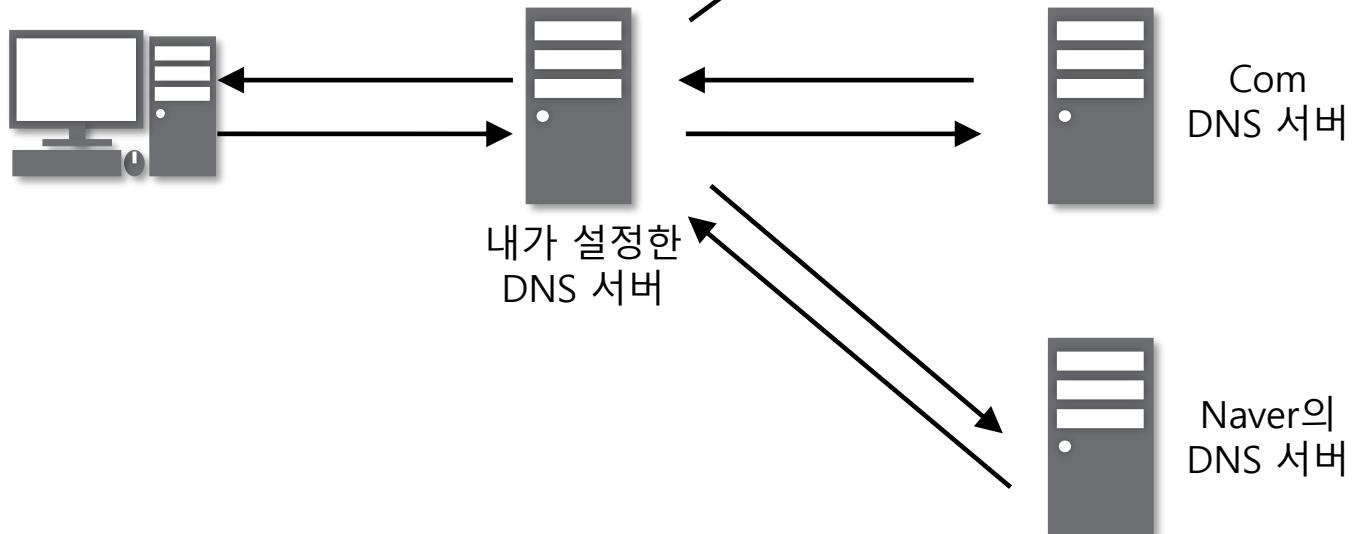
# UDP 프로토콜을 사용하는 프로그램

UDP 프로토콜을 사용하는 대표적인 프로그램들

〃

도메인을 물으면 IP를 알려주는  
DNS 서버

www.naver.com의 IP 주소는??



〃

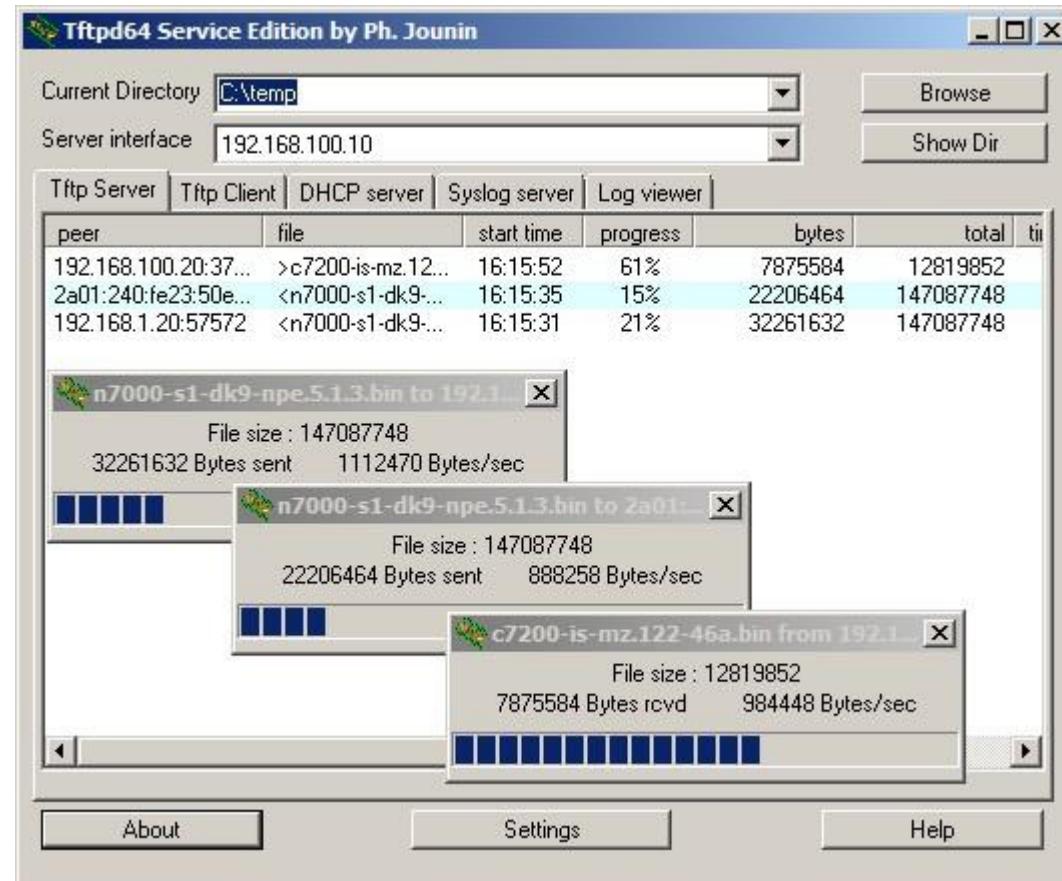
# UDP 프로토콜을 사용하는 프로그램

UDP 프로토콜을 사용하는 대표적인 프로그램들

//

UDP로 파일을 공유하는  
tftp 서버

//



<https://tftpd64.software.informer.com/download/>

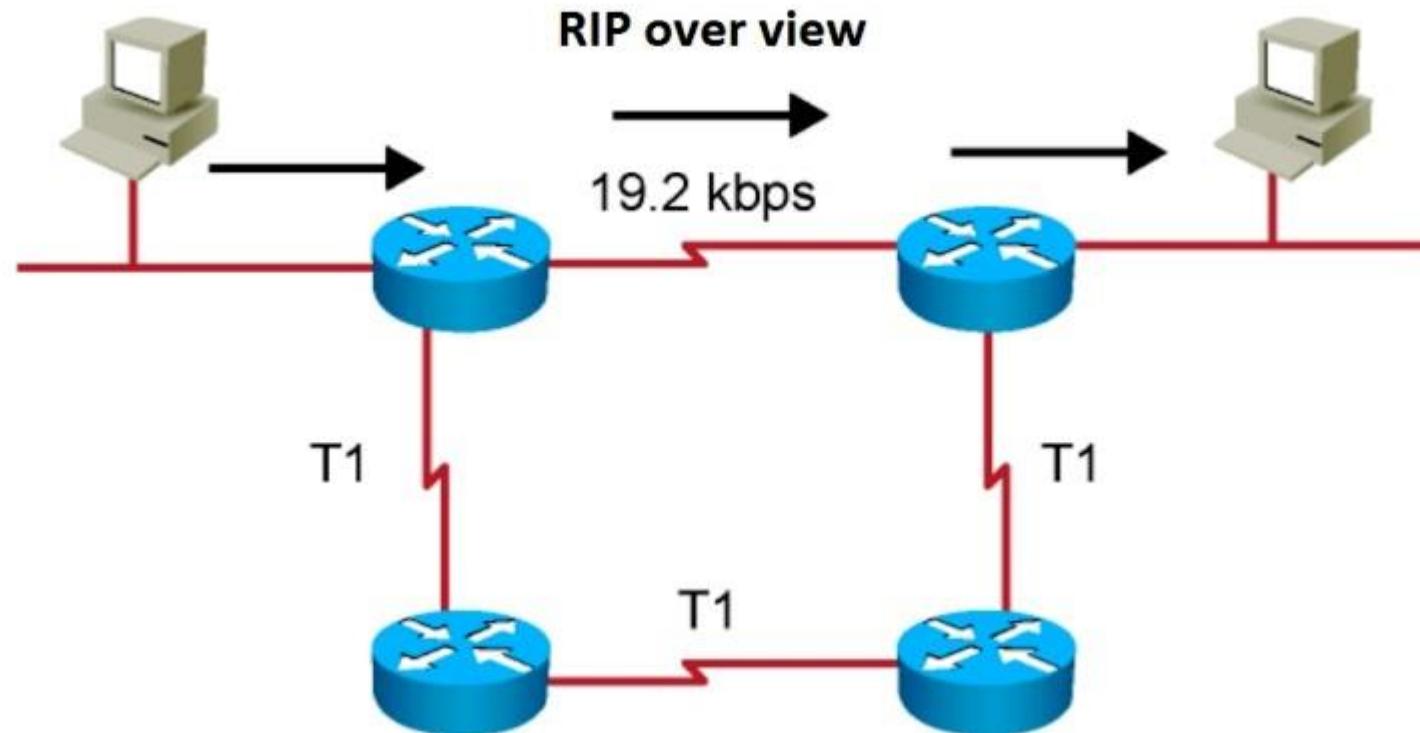
# UDP 프로토콜을 사용하는 프로그램

UDP 프로토콜을 사용하는 대표적인 프로그램들

〃

라우팅 정보를 공유하는  
RIP 프로토콜

〃



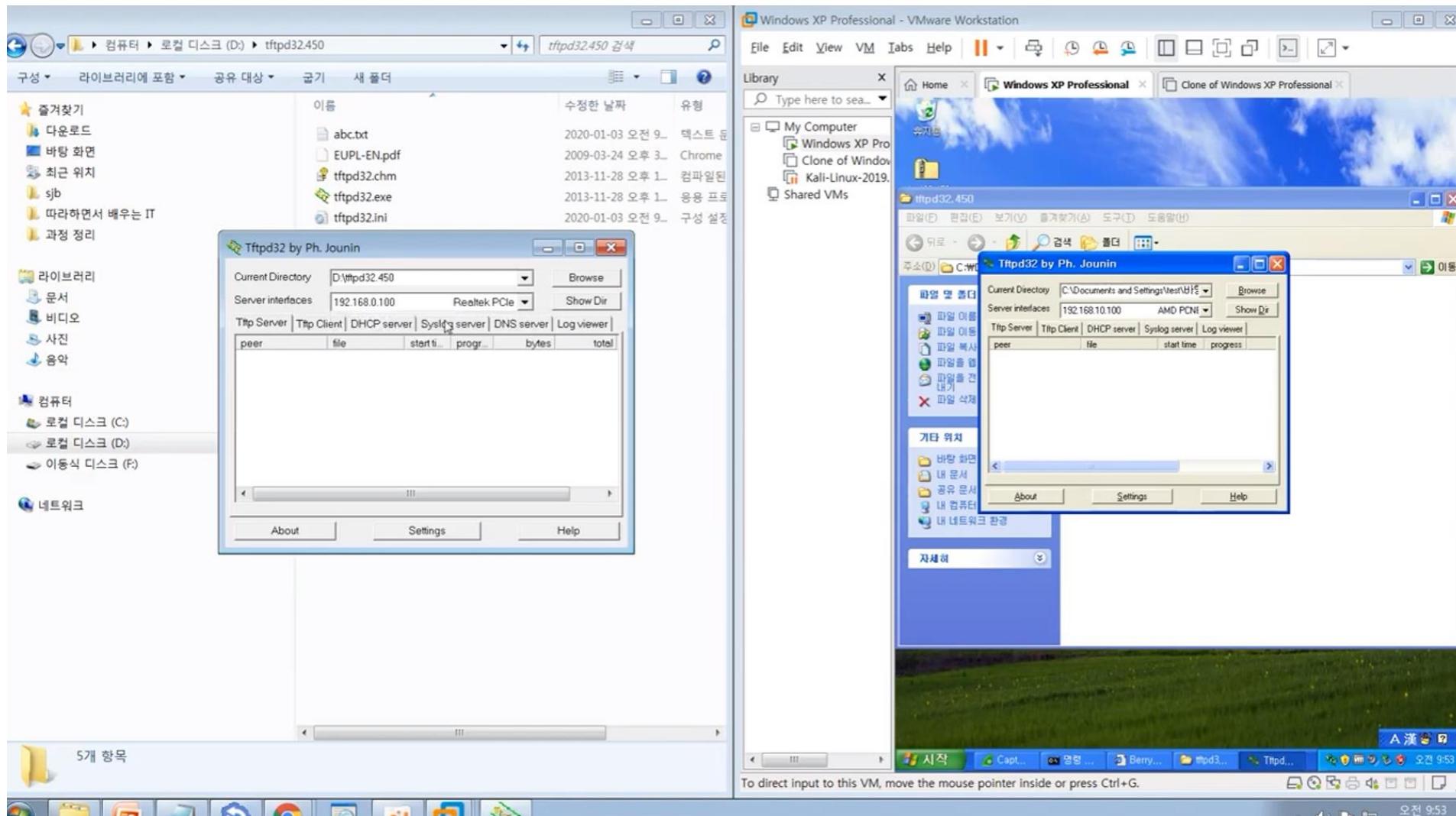
실습

## 1. tftpd 를 사용하여 데이터 공유해보기

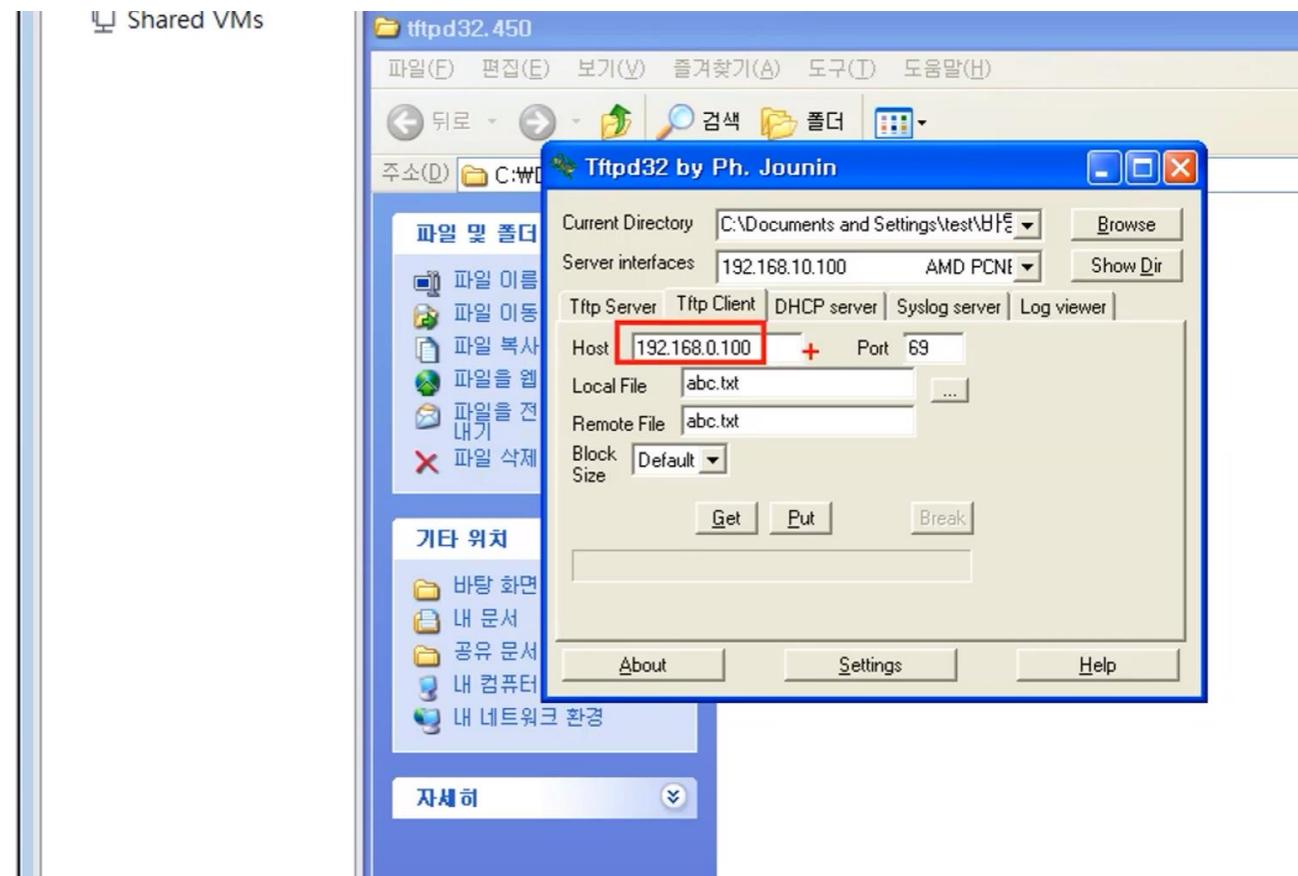
Tftpd 프로그램을 이용하여 UDP를 이용한 데이터 통신 해보기

## 2. 패킷 캡쳐 및 분석해보기

UDP패킷을 캡쳐해보고 분석해보기



# client



# **연결지향형 TCP 프로토콜**

# 목차

## INDEX



# **TCP 프로토콜**

# TCP 프로토콜

## TCP가 하는 일

전송 제어 프로토콜(Transmission Control Protocol, TCP)은 인터넷에 연결된 컴퓨터에서 실행되는 프로그램 간에 통신을 **안정적으로, 순서대로, 에러없이** 교환할 수 있게 한다.

TCP의 안정성을 필요로 하지 않는 애플리케이션의 경우 일반적으로 TCP 대신 비접속형 사용자 데이터그램 프로토콜(User Datagram Protocol)을 사용한다.

TCP는 UDP보다 안전하지만 느리다.

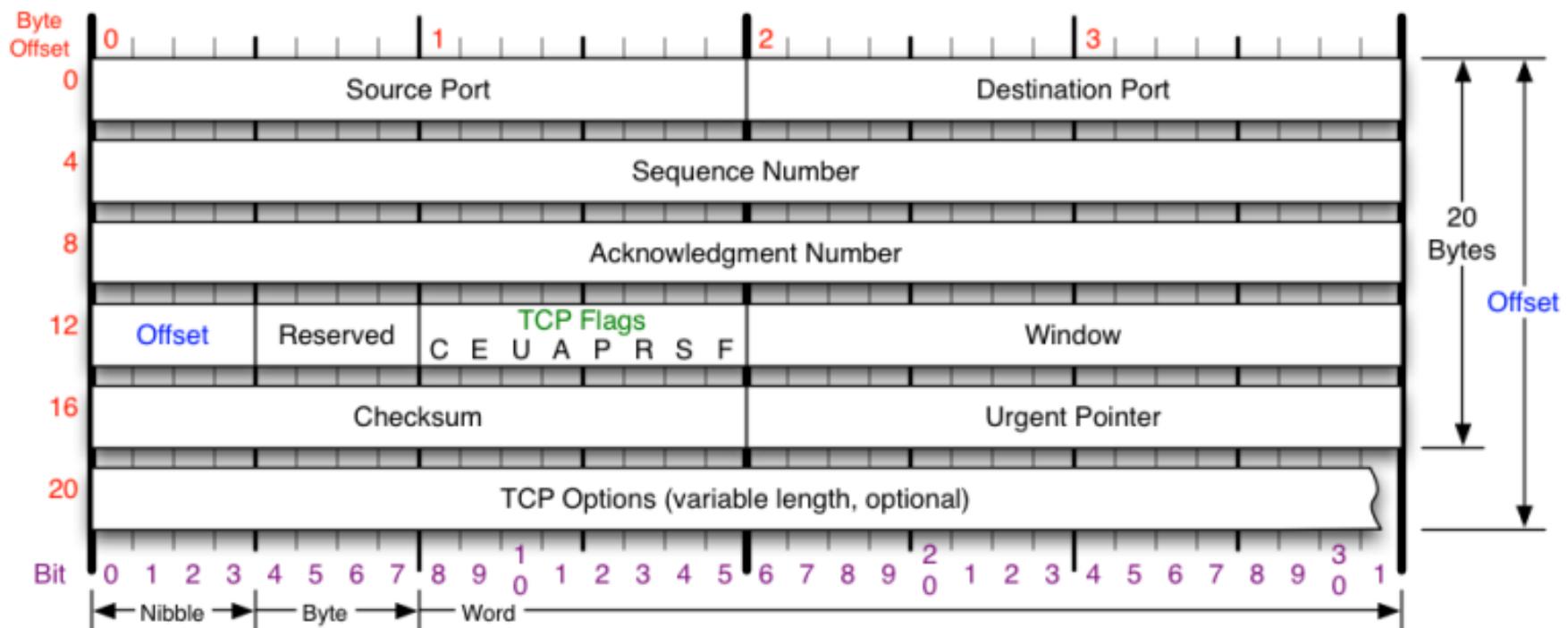
# TCP 프로토콜

## TCP 프로토콜의 구조

//

안전한 연결을 지향하는  
TCP 프로토콜

//



**TCP 플래그**

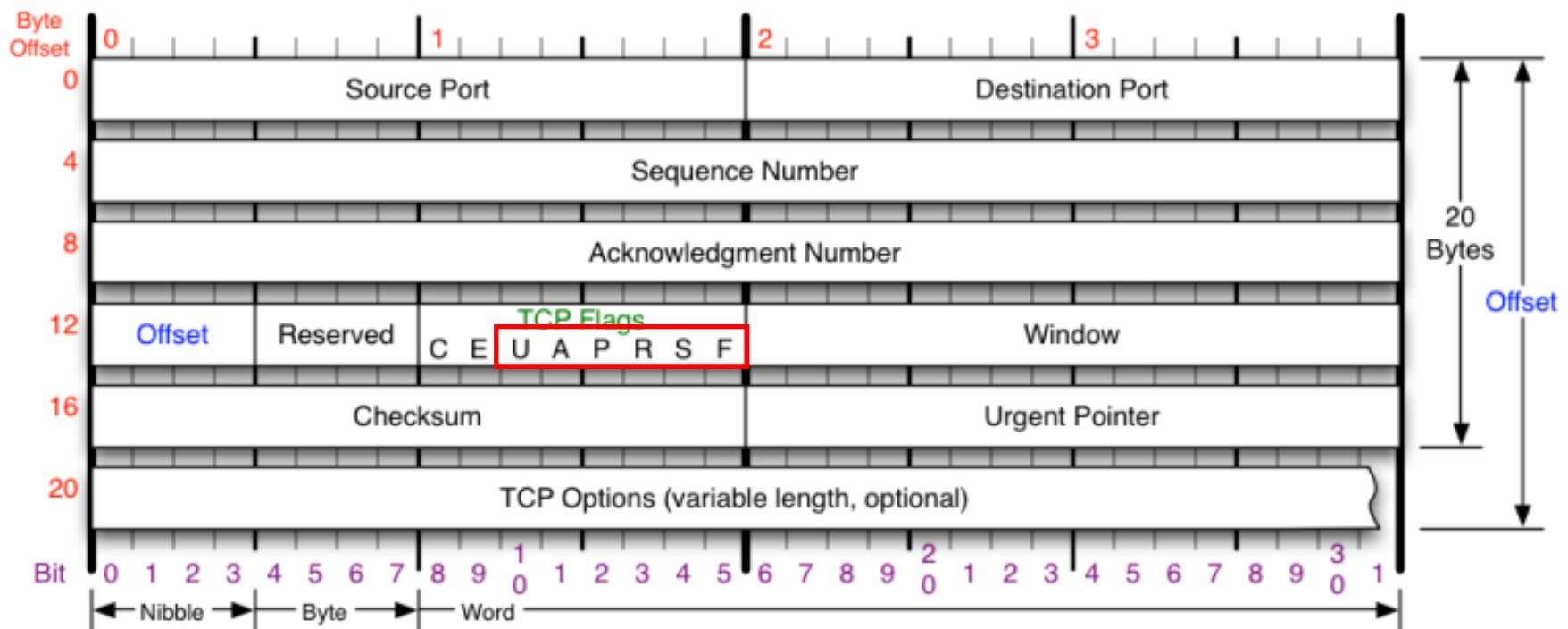
# TCP 플래그

## TCP 플래그의 종류

//

TCP 플래그

//



# TCP 플래그

## TCP 플래그의 종류

〃

TCP 플래그

〃

```
Wireshark - Packet 44 · 토컬 영역 연결
Flags: 0x002 (SYN)
000. .... .... = Reserved: Not set
...0 .... .... =Nonce: Not set
.... 0.... .... = Congestion Window Reduced (CWR): Not set
.... .0.. .... = ECN-Echo: Not set
.... ..0. .... = Urgent: Not set
.... ...0 .... = Acknowledgment: Not set
.... .... 0.... = Push: Not set
.... .... .0.. = Reset: Not set
D .... .... .1. = Syn: Set
.... .... ....0 = Fin: Not set
[TCP Flags: .....S.]
```

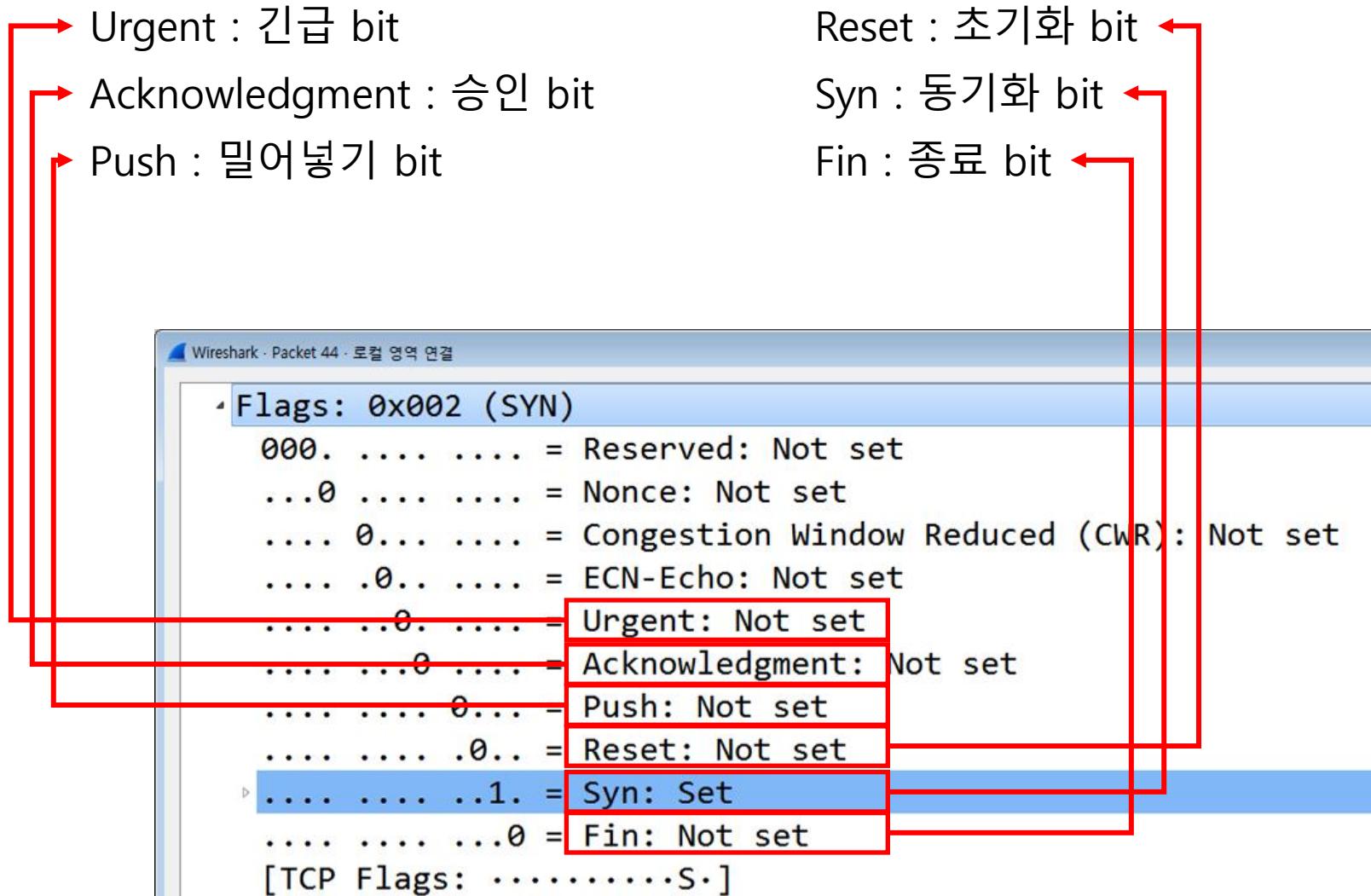
# TCP 플래그

## 각 플래그의 기능

〃

TCP 플래그

〃



# TCP를 이용한 통신과정

# TCP를 이용한 통신과정

## 연결 수립 과정

TCP를 이용한 데이터 통신을 할 때 프로세스와 프로세스를 연결하기 위해  
가장 먼저 수행되는 과정

1. 클라이언트가 서버에게 요청 패킷을 보내고
2. 서버가 클라이언트의 요청을 받아들이는 패킷을 보내고
3. 클라이언트는 이를 최종적으로 수락하는 패킷을 보낸다.

위의 3개의 과정을 3Way Handshake라고 부른다.

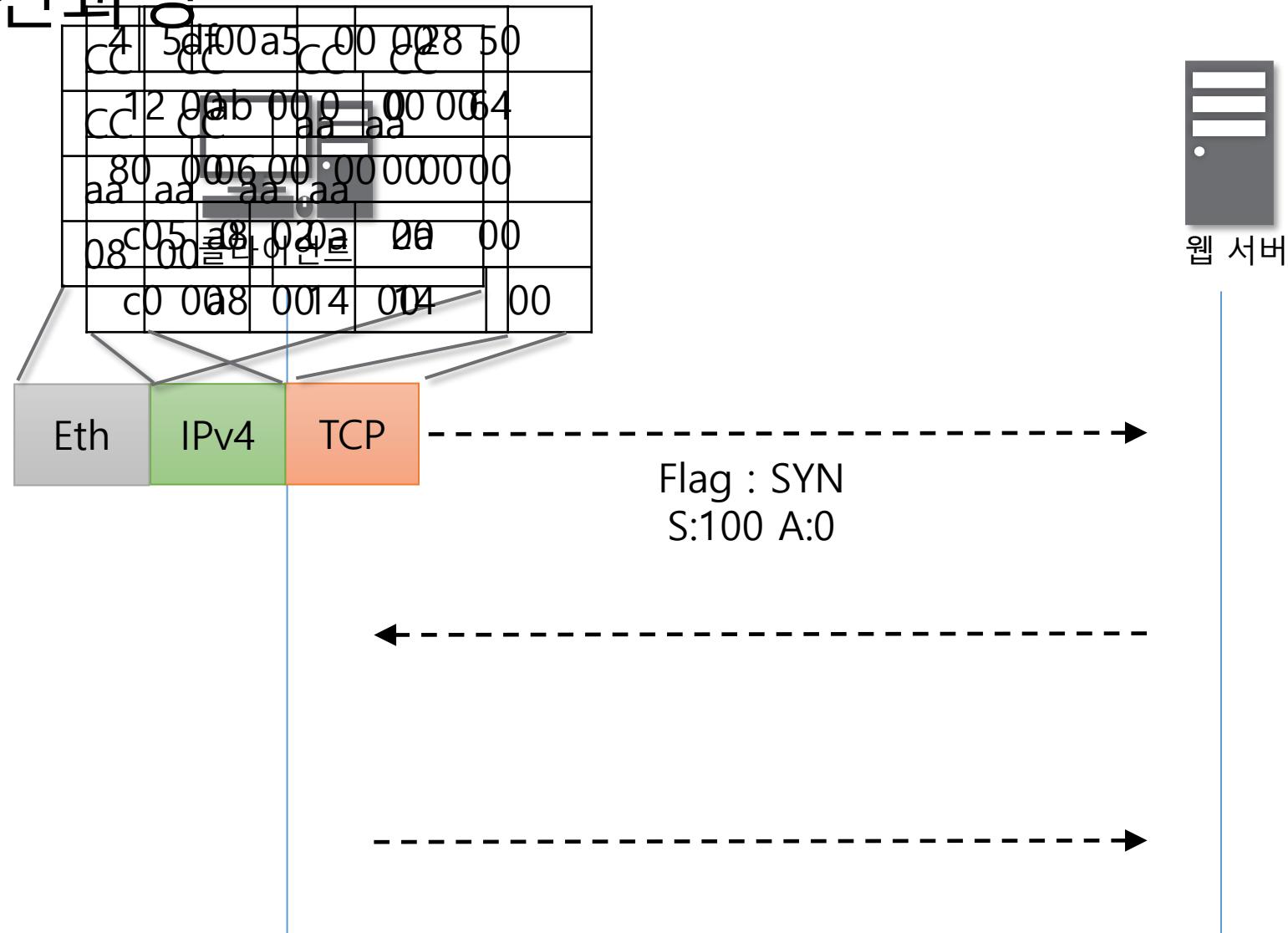
# TCP를 이용한 통신 과정

연결 수립 과정

〃

연결 수립을 하기 위한 통신  
TCP 3Way Handshake

〃



# TCP를 이용한 통신과정

연결 수립 과정

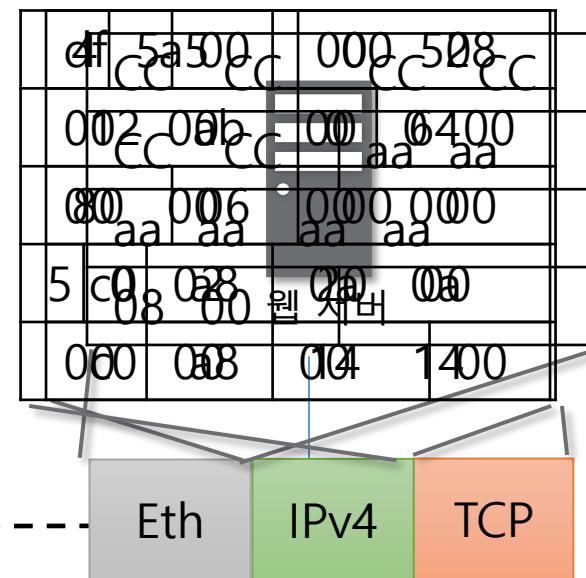


//

연결 수립을 하기 위한 통신  
TCP 3Way Handshake

//

Flag : SYN  
S:100 A:0



# TCP를 이용한 통신과정

## 연결 수립 과정

〃

연결 수립을 하기 위한 통신  
TCP 3Way Handshake

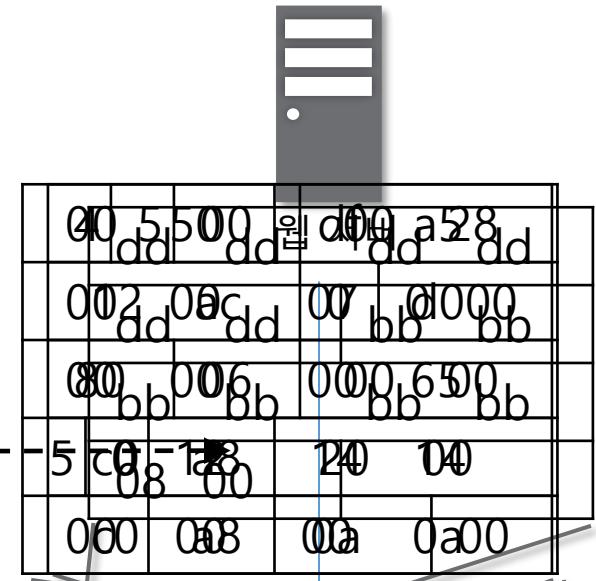
〃



클라이언트

Flag : SYN  
S:100 A:0

Flag : SYN+ACK  
S:2000 A:101



Eth IPv4 TCP

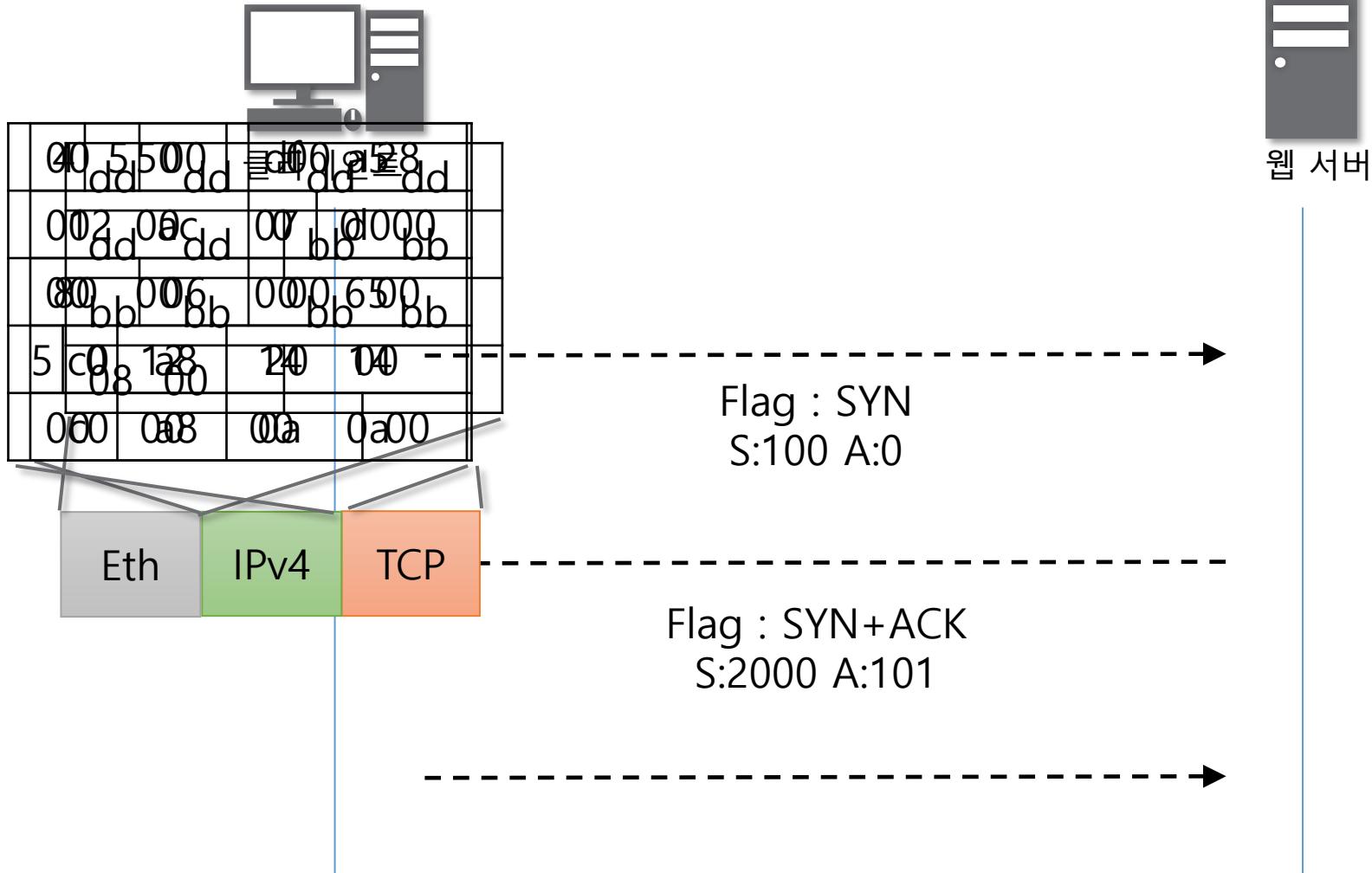
# TCP를 이용한 통신과정

연결 수립 과정

〃

연결 수립을 하기 위한 통신  
TCP 3Way Handshake

〃



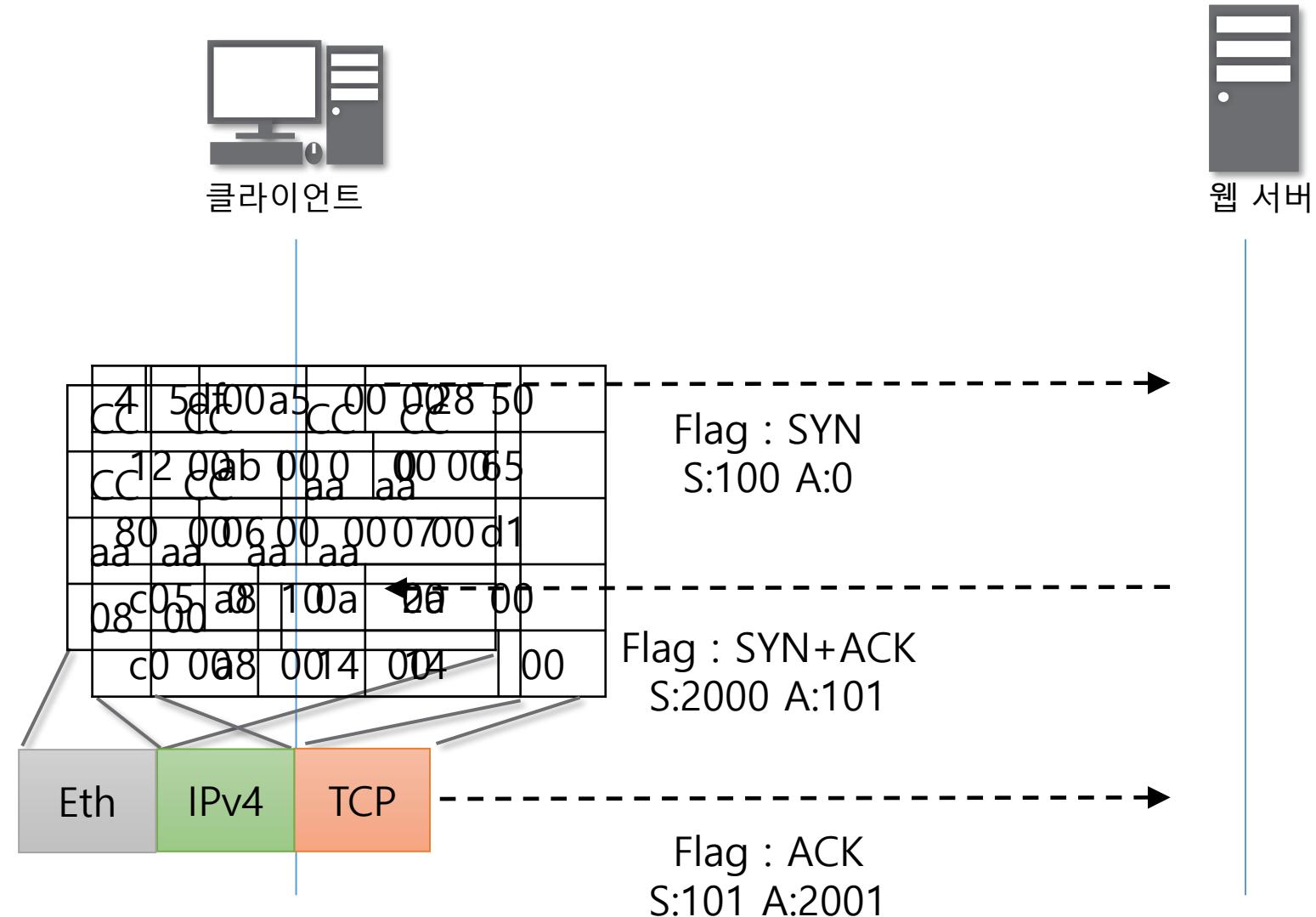
# TCP를 이용한 통신과정

연결 수립 과정

〃

연결 수립을 하기 위한 통신  
TCP 3Way Handshake

〃



# TCP를 이용한 통신과정

## 데이터 송수신 과정

TCP를 이용한 데이터 통신을 할 때 단순히 TCP 패킷만을 캡슐화해서 통신하는 것이 아닌 페이로드를 포함한 패킷을 주고 받을 때의 일정한 규칙

1. 보낸 쪽에서 또 보낼 때는 SEQ번호와 ACK번호가 그대로다.
2. 받는 쪽에서 SEQ번호는 받은 ACK번호가 된다.
3. 받는 쪽에서 ACK번호는 받은 SEQ번호 + **데이터의 크기**

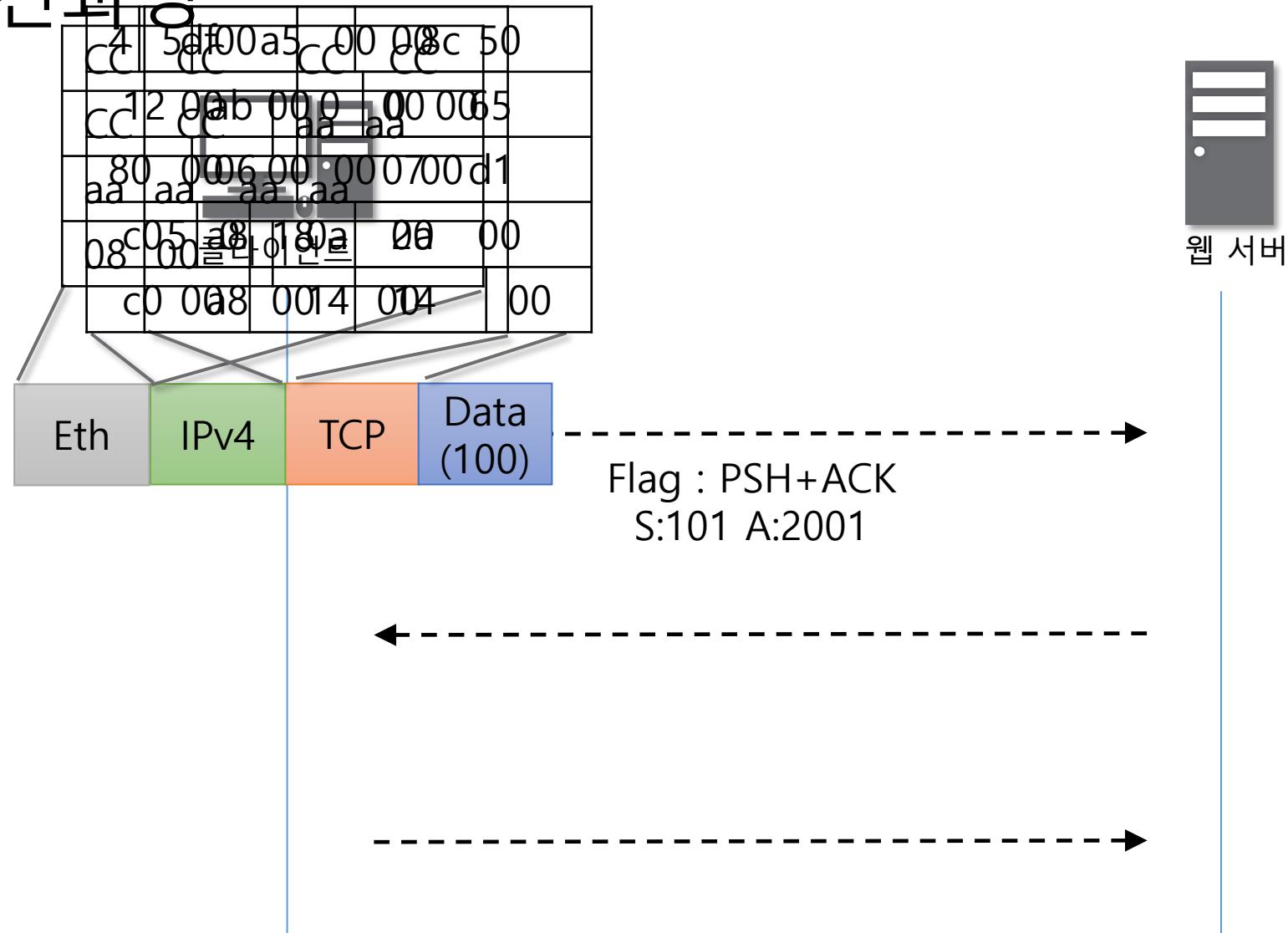
# TCP를 이용한 통신 과정

데이터 송수신 과정

〃

HTTP나 FTP와 같은 각종  
데이터를 포함한 통신

〃



## TCP를 이용한 통신과정

# 데이터 송수신 과정



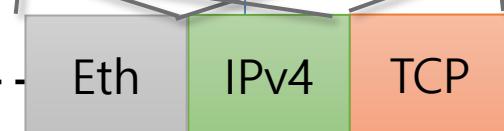
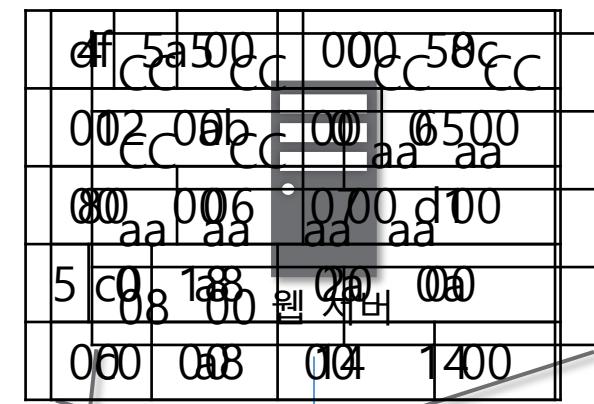
## 클라이언트

11

HTTP나 FTP와 같은 각종  
데이터를 포함한 통신

11

Flag : PSH+ACK  
S:101 A:2001



# TCP를 이용한 통신과정

## 데이터 송수신 과정

〃

HTTP나 FTP와 같은 각종  
데이터를 포함한 통신

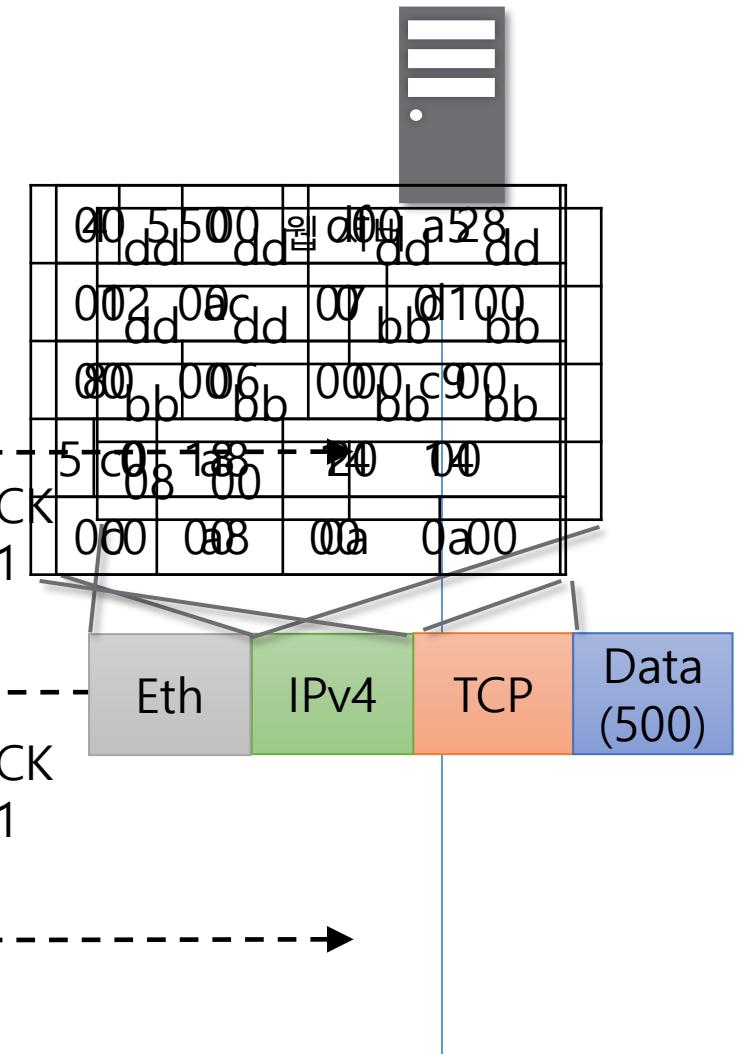
〃



클라이언트

Flag : PSH+ACK  
S:101 A:2001

Flag : PSH+ACK  
S:2001 A:201



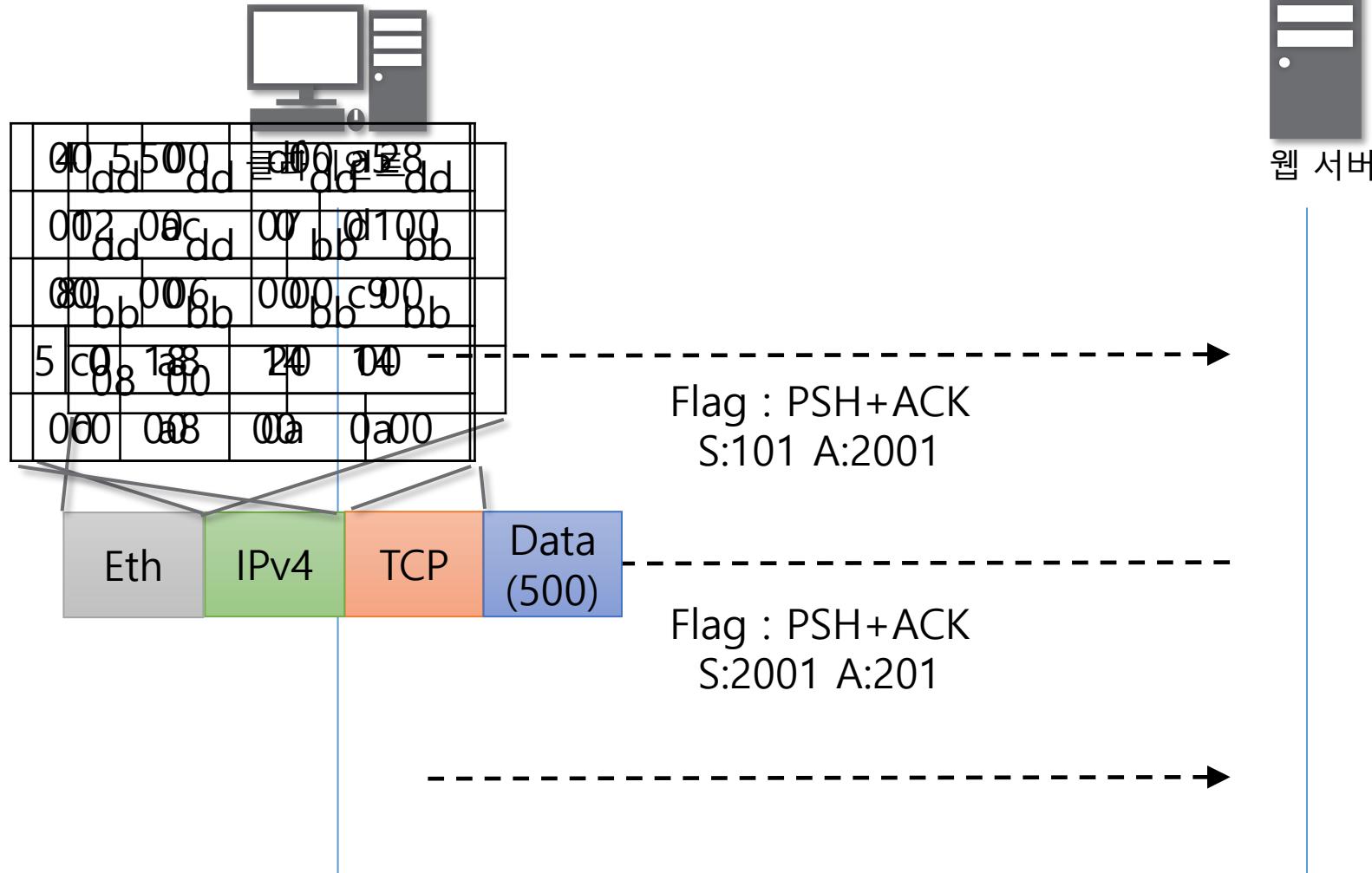
# TCP를 이용한 통신과정

데이터 송수신 과정

〃

HTTP나 FTP와 같은 각종  
데이터를 포함한 통신

〃

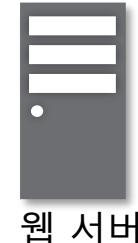


# TCP를 이용한 통신과정

데이터 송수신 과정



클라이언트

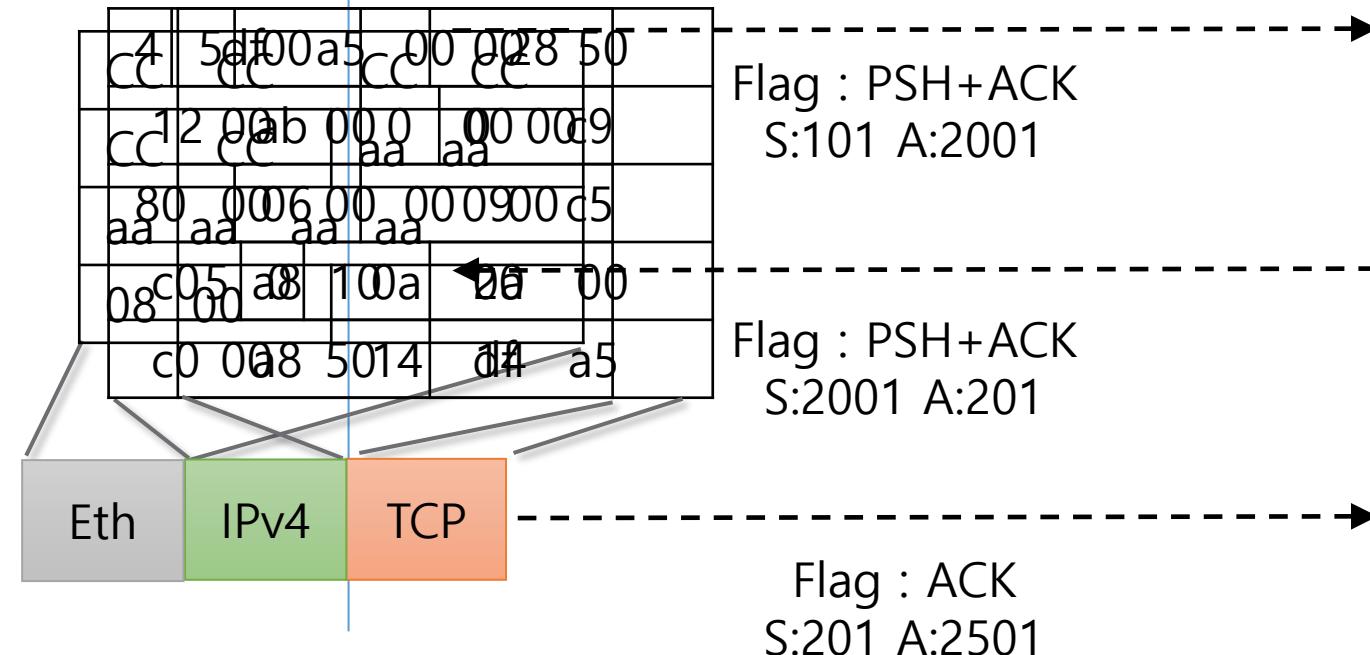


웹 서버

〃

HTTP나 FTP와 같은 각종  
데이터를 포함한 통신

〃



**TCP 상태전이도**

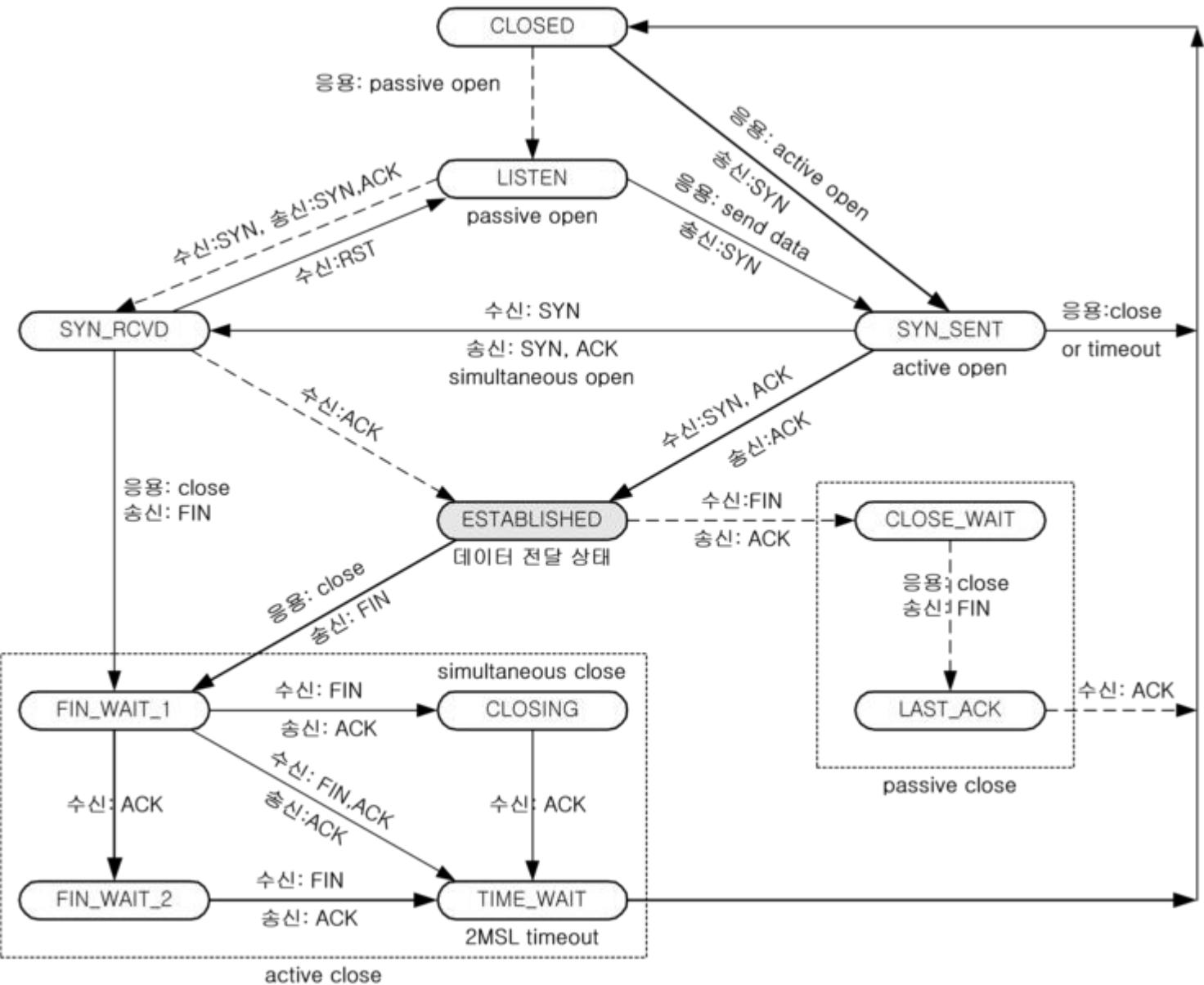
# TCP 상태전이도

## TCP 연결 상태의 변화

//

TCP의 여러가지  
상태 변화

//



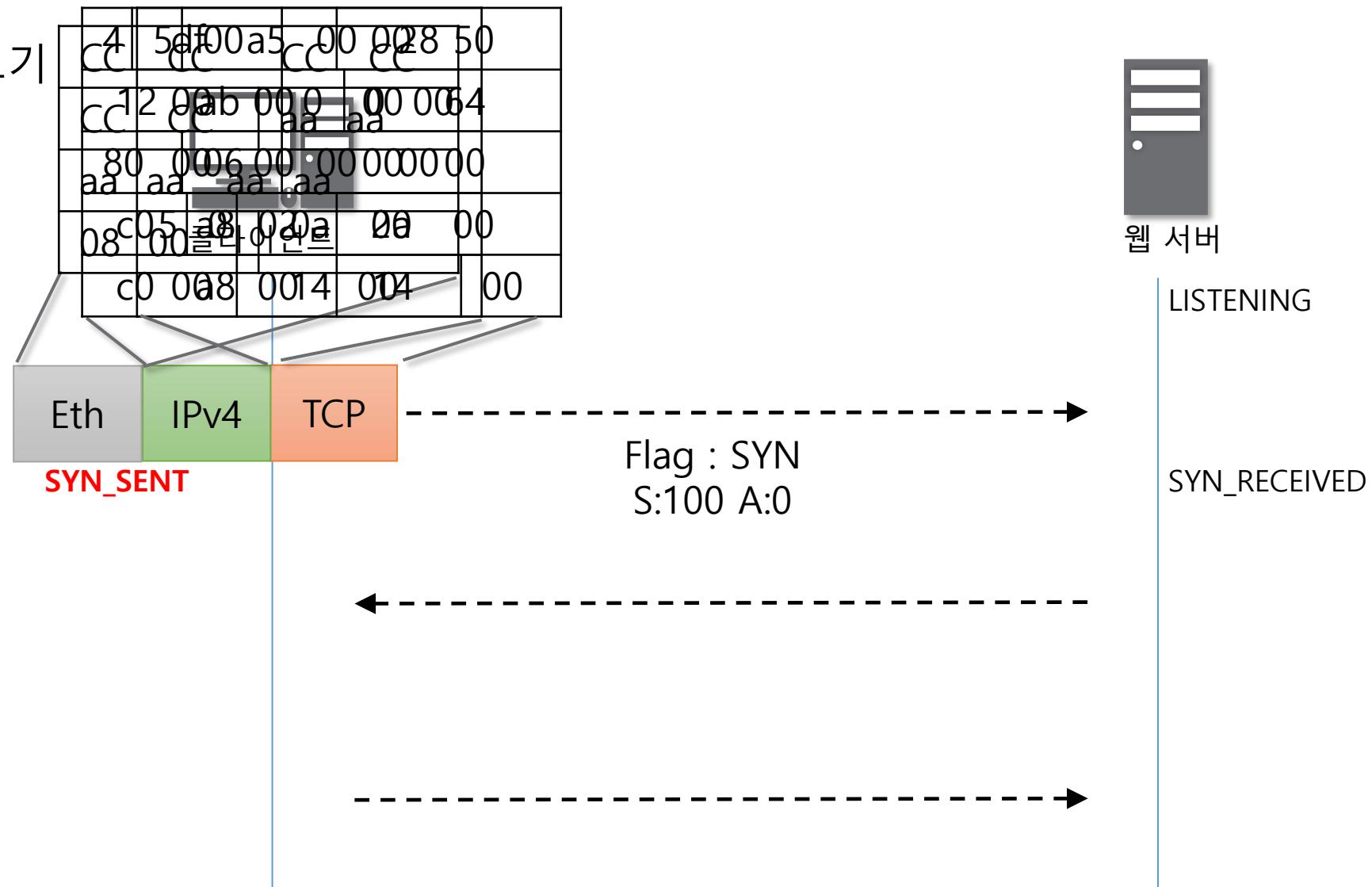
# TCP 상태전이도

3Way-Handshake와 함께보기

//

연결을 수립하는  
3Way-Handshake 과정  
에서의 상태 변화

//



# TCP 상태전이도

3Way-Handshake와 함께보기



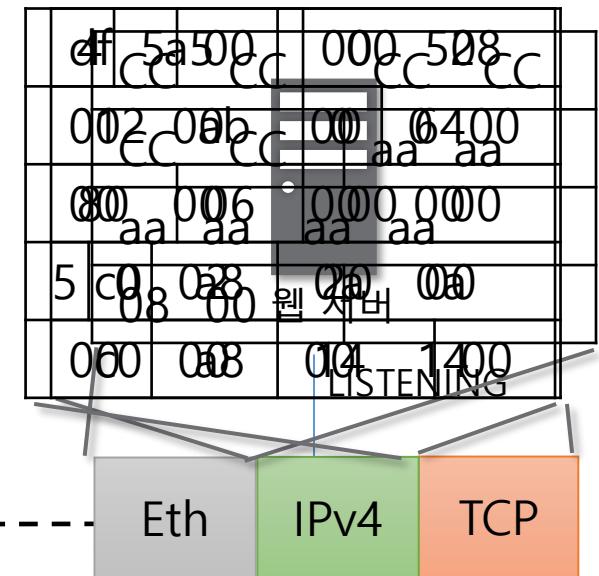
//

연결을 수립하는  
3Way-Handshake 과정  
에서의 상태 변화

//

SYN\_SENT

Flag : SYN  
S:100 A:0



SYN\_RECEIVED

# TCP 상태전이도

3Way-Handshake와 함께보기

〃

연결을 수립하는  
3Way-Handshake 과정  
에서의 상태 변화

〃

SYN\_SENT

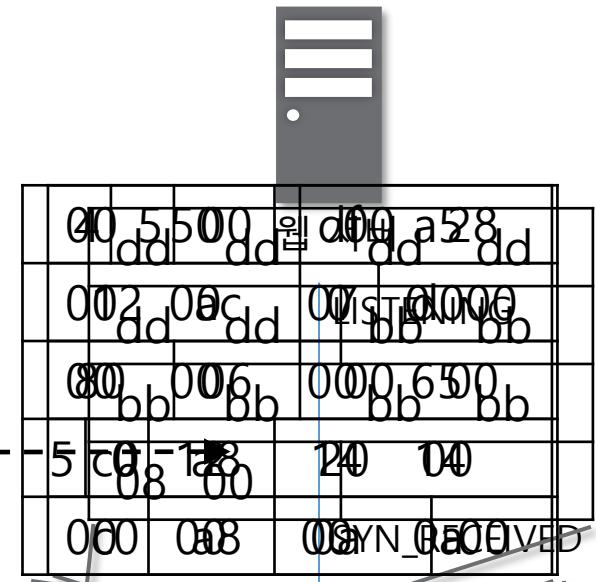
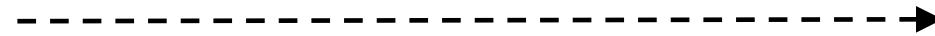


클라이언트

Flag : SYN  
S:100 A:0



Flag : SYN+ACK  
S:2000 A:101



Eth IPv4 TCP

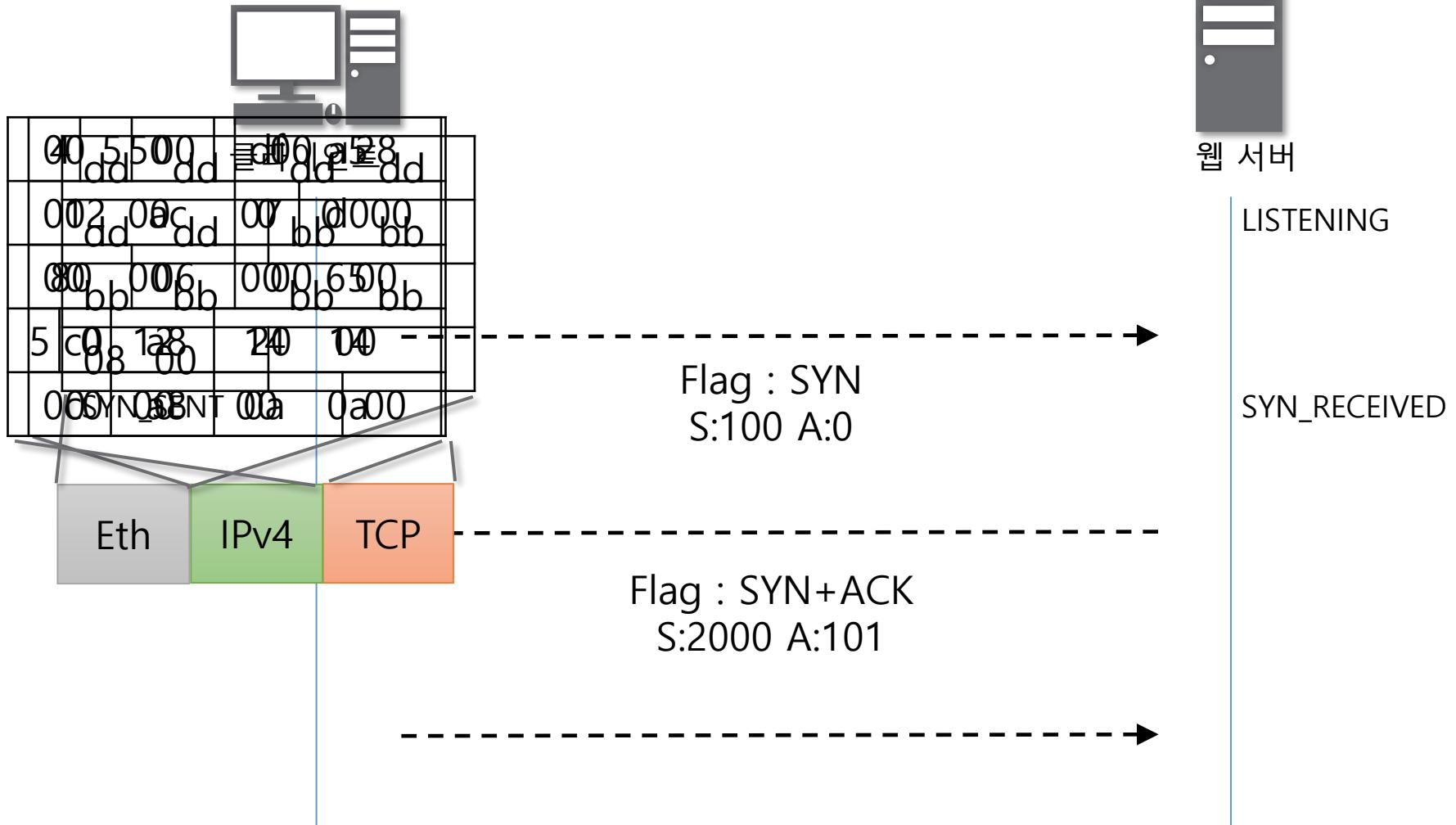
# TCP 상태전이도

3Way-Handshake와 함께보기

〃

연결을 수립하는  
3Way-Handshake 과정  
에서의 상태 변화

〃



# TCP 상태전이도

3Way-Handshake와 함께보기



클라이언트

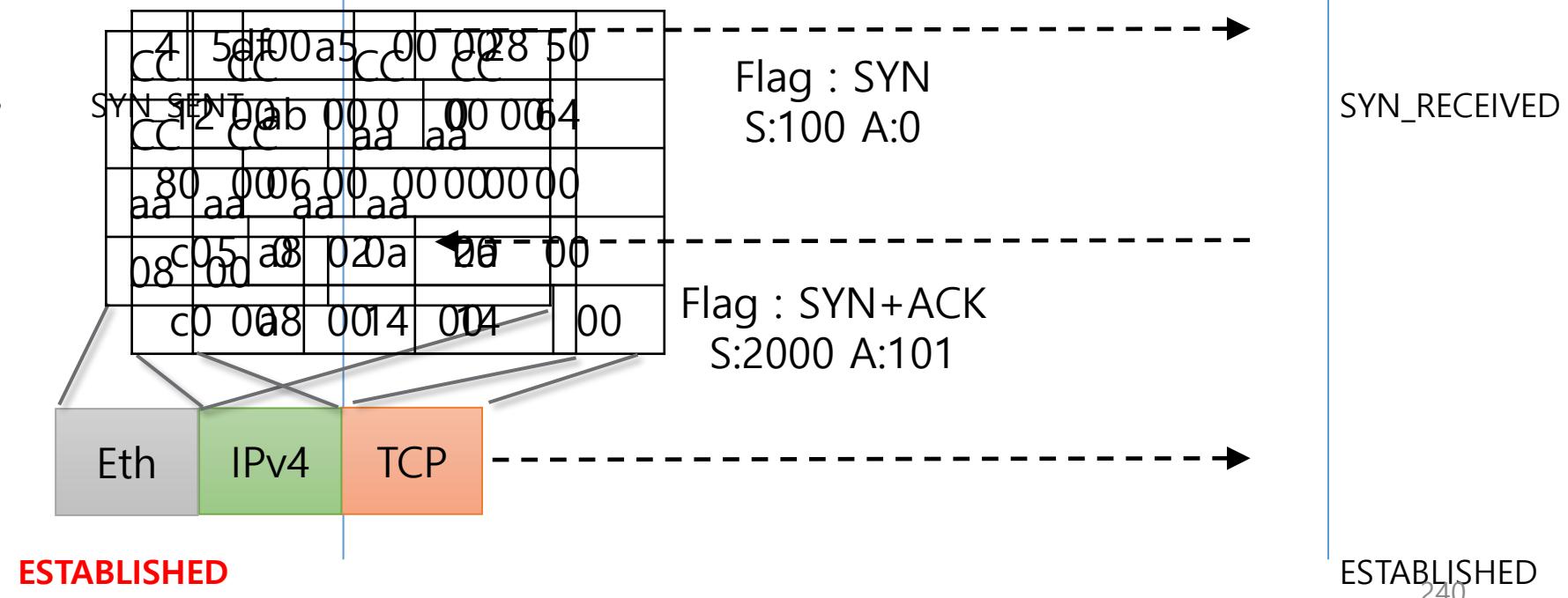


웹 서버

〃

연결을 수립하는  
3Way-Handshake 과정  
에서의 상태 변화

〃



실습

## 1. TCP 3Way Handshake 과정 계산해보기

TCP 3Way Handshake 과정에서  
플래그와 Seq번호, Ack번호를 확인해가며 직접 계산해보기

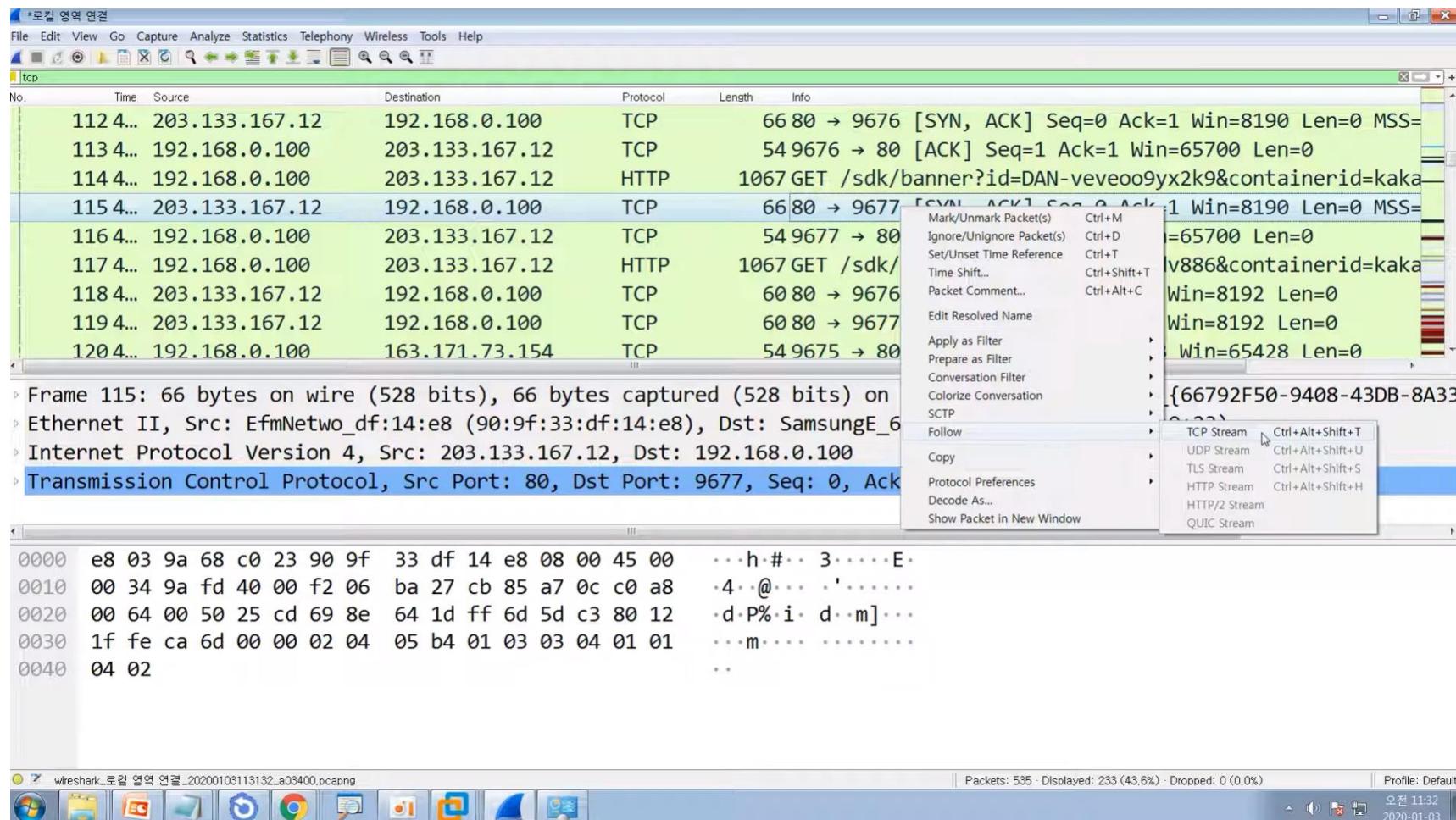
## 2. TCP 프로토콜 분석하기

TCP를 이용한 통신 과정을 Wireshark로 캡쳐하여 해당 패킷을 분석해보기

## 3. 데이터 송수신 과정 계산해보기

TCP를 이용한 통신을 할 때 데이터를 주고 받는 과정에서  
플래그와 Seq번호, Ack번호를 확인해가며 직접 계산해보기

# TCP Stream



# 목차

## INDEX

### NAT

NAT란?

### 포트포워딩

포트포워딩이  
란?

### 실습

포트포워딩 설정해보기  
사설 IP를 사용하는  
서버로 접속해보기

# NAT

# NAT와 포트포워딩

# NAT

## NAT란?

NAT(Network Address Translation)은 IP 패킷의 TCP/UDP 포트 숫자와 소스 및 목적지의 IP 주소 등을 재기록하면서 라우터를 통해 네트워크 트래픽을 주고 받는 기술을 말한다.

패킷에 변화가 생기기 때문에 IP나 TCP/UDP의 체크섬(checksum)도 다시 계산되어 재기록해야 한다.

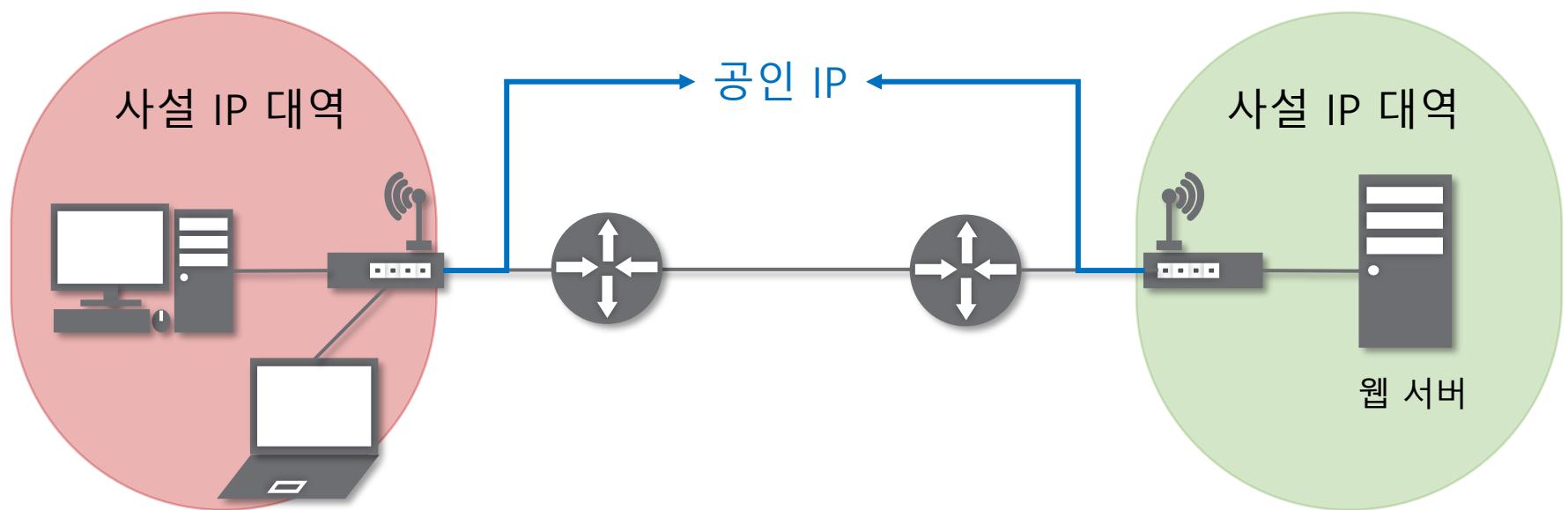
NAT를 이용하는 이유는 대개 사설 네트워크에 속한 여러 개의 호스트가 하나의 공인 IP 주소를 사용하여 인터넷에 접속하기 위함이다.

하지만 꼭 사설IP를 공인IP로 변환 하는 데에만 사용하는 기술은 아니다.

# NAT

NAT란?

"  
실제 일반적인 네트워크의 모습  
사설IP와 공인IP  
"



# NAT

NAT란?

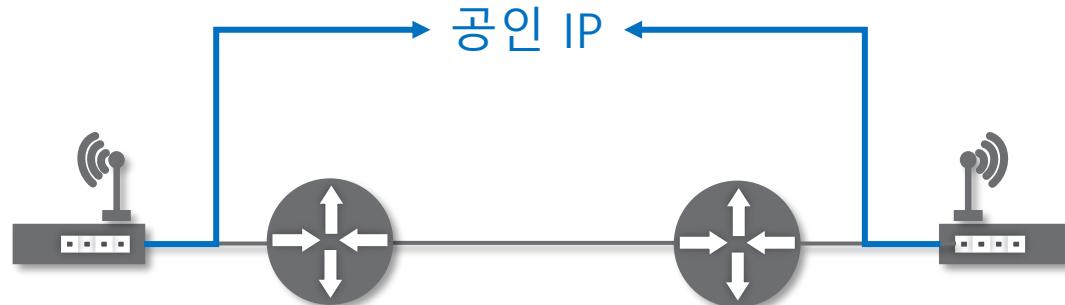
---

〃

인터넷 세상에서 바라본 모습  
사설IP와 공인IP

〃

---



**포트포워딩**

# 포트포워딩

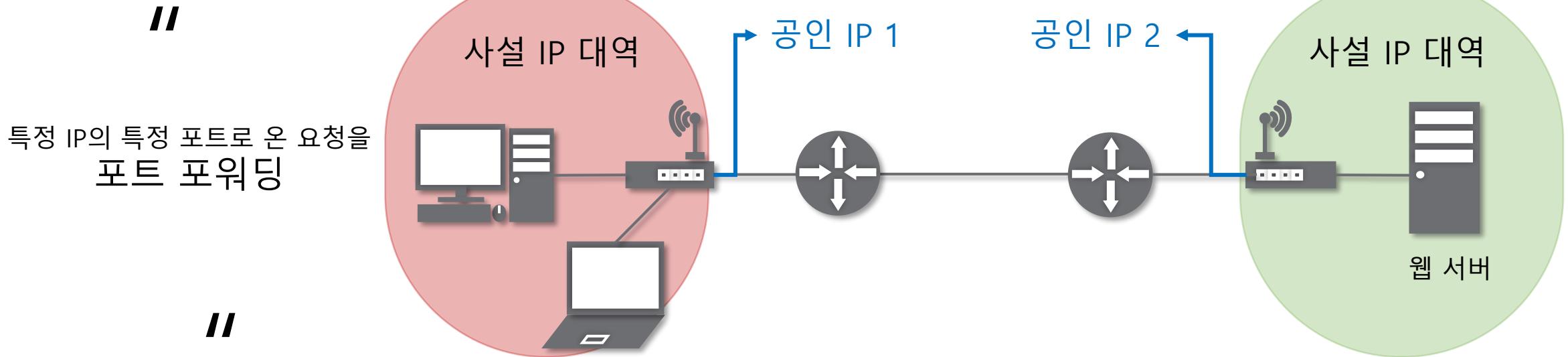
## 포트포워딩이란?

포트 포워딩 또는 포트 매핑(port mapping)은 패킷이 라우터나 방화벽과 같은 네트워크 장비를 가로지르는 동안 특정 IP 주소와 포트 번호의 통신 요청을 특정 다른 IP와 포트 번호로 넘겨주는 네트워크 주소 변환(NAT)의 응용이다.

이 기법은 게이트웨이(외부망)의 반대쪽에 위치한 사설네트워크에 상주하는 호스트에 대한 서비스를 생성하기 위해 흔히 사용된다.

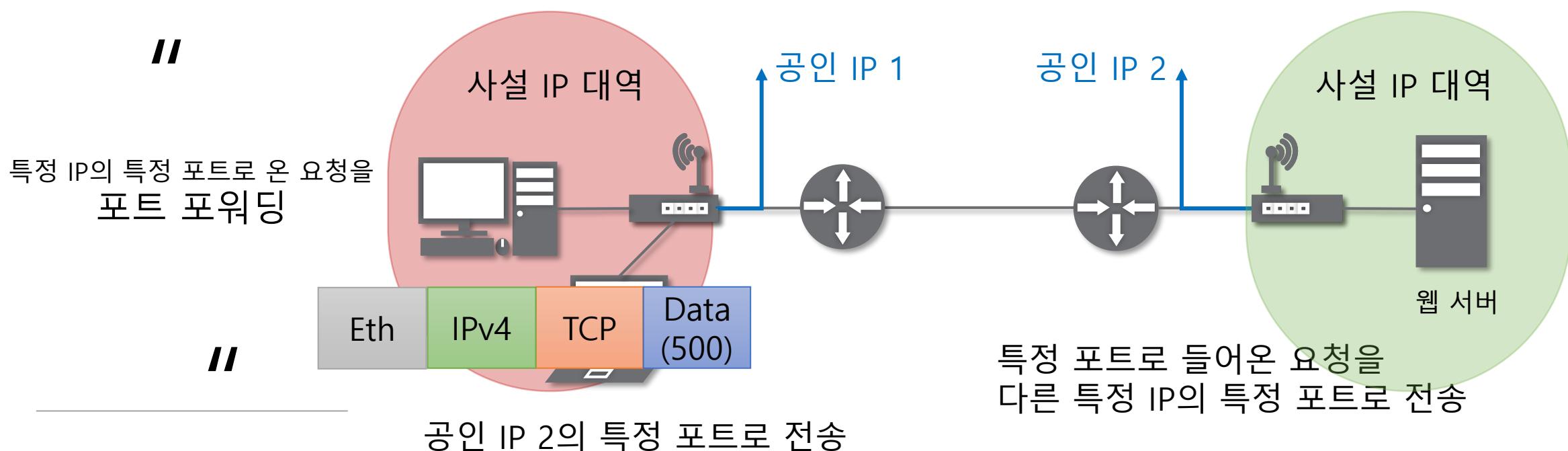
# 포트포워딩

## 포트포워딩이란?



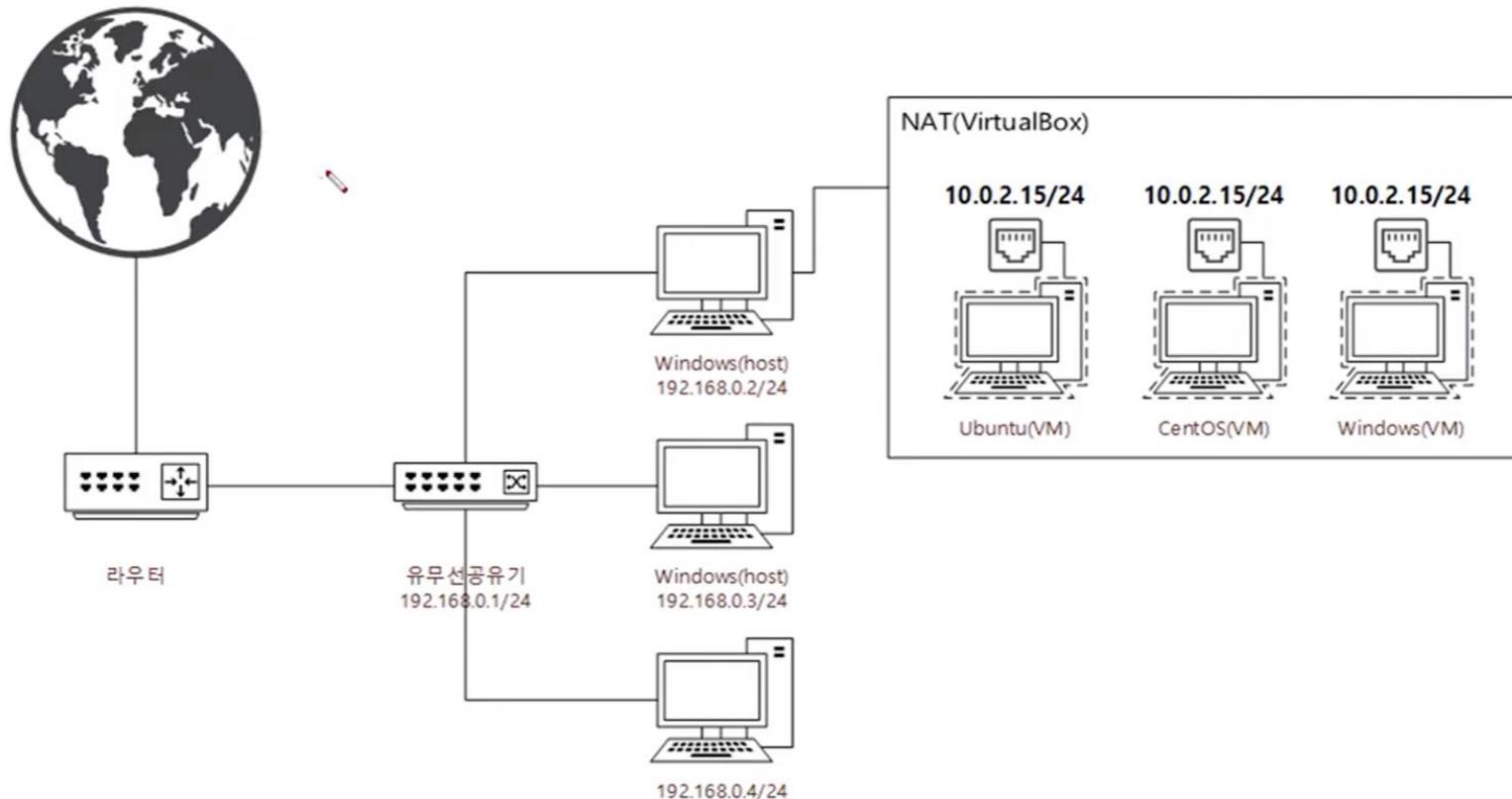
# 포트포워딩

## 포트포워딩이란?

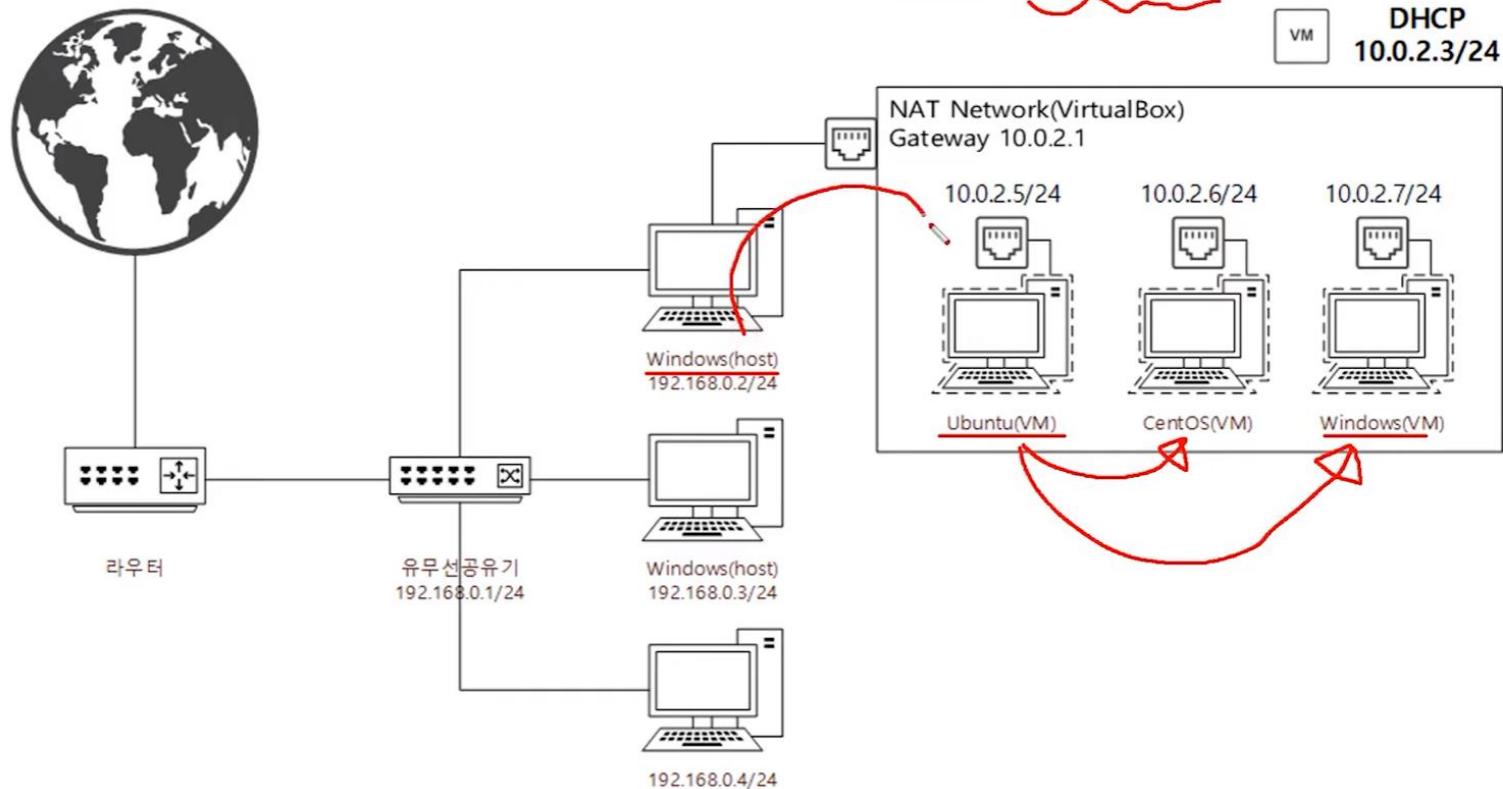


실습

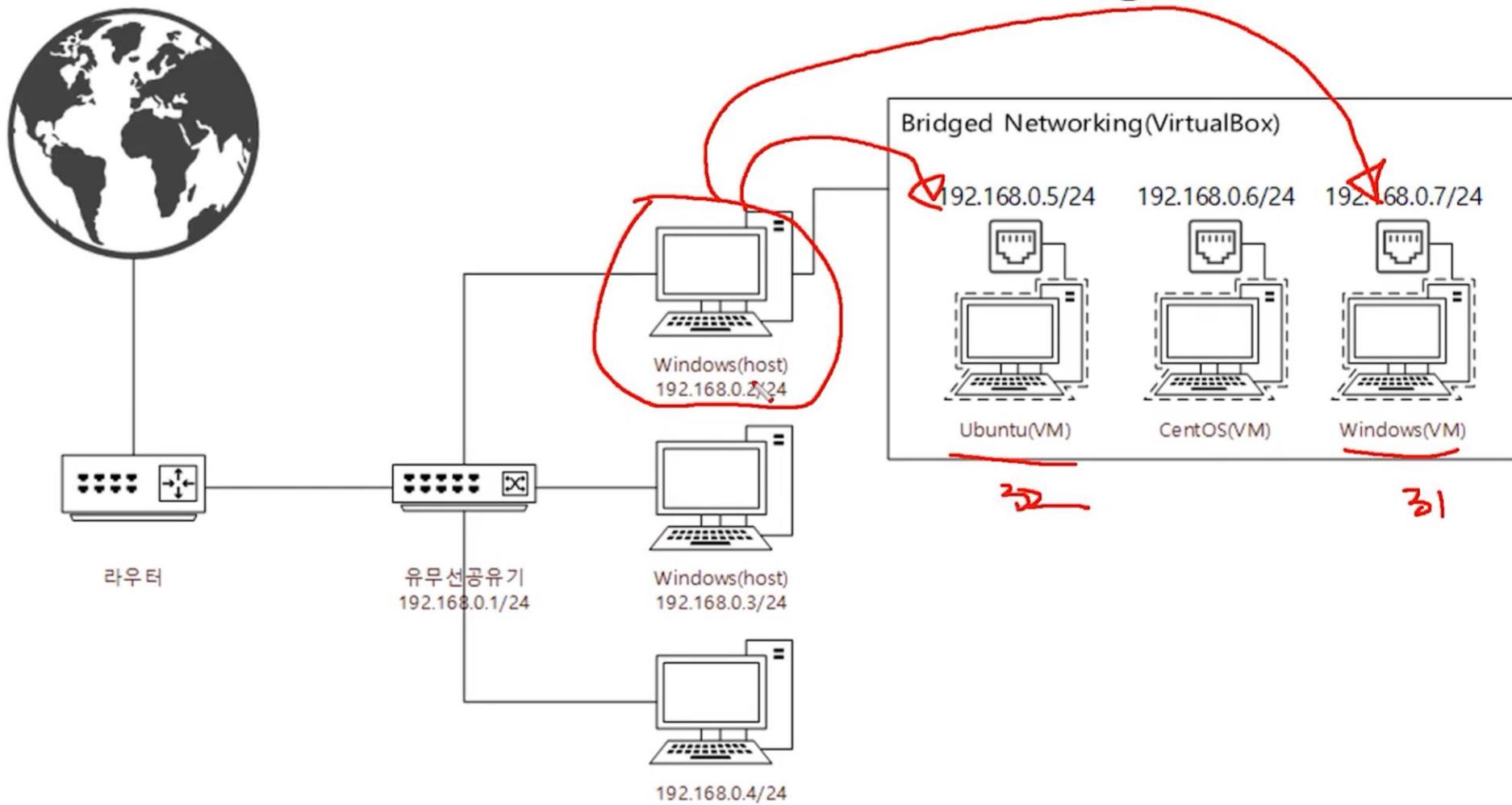
# VirtualBox Network (NAT)



# VirtualBox Network (NAT Network)



# VirtualBox Network (Bridge)



## 1. 포트포워딩 설정해보기

포트포워딩을 이용하여 다른 사용자들이 사설IP를 사용하는  
서버로 접속 할 수 있도록 설정해보기

## 2. 사설 IP를 사용하는 서버로 접속해보기

사설IP를 사용하는 가상머신에 서버를 설정하고 해당 서버를 포트포워딩을 통해 접속할 수  
있도록 설정해보기

# **7계층 프로토콜 HTTP**

# 목차

## INDEX

### HTTP 프로토콜

### HTTP 요청 프로토콜

### HTTP 응답 프로토콜

### HTTP 헤더 포맷

### 따라 學IT

웹을 만드는 기술들  
HTTP 프로토콜의 특징  
HTTP 프로토콜의  
통신 과정

HTTP 요청  
프로토콜의 구조  
요청 타입  
URI

HTTP 응답  
프로토콜의 구조  
상태 코드

HTTP 헤더 구조  
일반 헤더  
요청 헤더  
응답 헤더

HTTP 작성 실습  
HTTP 수정 실습

# HTTP 프로토콜

# HTTP 프로토콜

웹을 만드는 기술들

//

웹을 만들기 위해 사용되는  
다양한 기술들

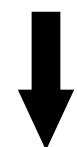
//

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수

- 
- Python
  - Spring
  - Jquery
  - Ajax



선택

# HTTP 프로토콜

웹을 만드는 기술들

“

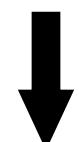
웹을 만들기 위해 사용되는  
다양한 기술들

“

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수



선택

- Python
- Spring
- Jquery
- Ajax

# HTTP 프로토콜

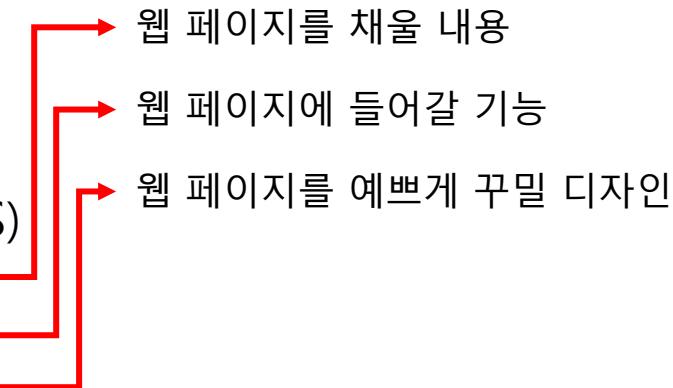
웹을 만드는 기술들

“

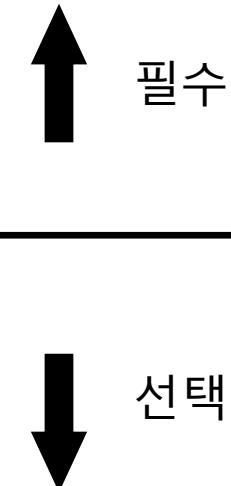
웹을 만들기 위해 사용되는  
다양한 기술들

”

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



- Python
- Spring
- Jquery
- Ajax



# HTTP 프로토콜

웹을 만드는 기술들

“

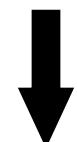
웹을 만들기 위해 사용되는  
다양한 기술들

“

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수



선택

- Python
- Spring
- Jquery
- Ajax

# HTTP 프로토콜

웹을 만드는 기술들

“

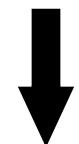
웹을 만들기 위해 사용되는  
다양한 기술들

“

- **HTTP** (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수



선택

- Python
- Spring
- Jquery
- Ajax

# HTTP 프로토콜

웹을 만드는 기술들

---

〃

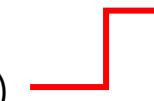
웹을 만들기 위해 사용되는  
다양한 기술들

---

〃

- **HTTP** (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB

HTML과 JS와 CSS같은 파일을  
웹 서버에게 요청하고 받아오는  
프로토콜

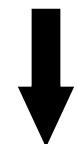


필수



- Python
- Spring
- Jquery
- Ajax

선택



# HTTP 프로토콜

웹을 만드는 기술들

“

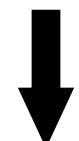
웹을 만들기 위해 사용되는  
다양한 기술들

“

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수



선택

- Python
- Spring
- Jquery
- Ajax

# HTTP 프로토콜

웹을 만드는 기술들

“

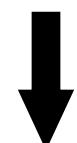
웹을 만들기 위해 사용되는  
다양한 기술들

“

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수



선택

- Python
- Spring
- Jquery
- Ajax

# HTTP 프로토콜

웹을 만드는 기술들

“

웹을 만들기 위해 사용되는  
다양한 기술들

”

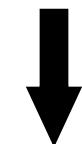
- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB

웹 서버 페이지를 만드는 기술들



필수

- Python
- Spring
- Jquery
- Ajax



선택

# HTTP 프로토콜

웹을 만드는 기술들

“

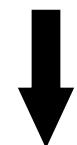
웹을 만들기 위해 사용되는  
다양한 기술들

“

- HTTP (HTTPS -> SSL/TLS)
- HTML
- Javascript
- CSS
- ASP/ASP.NET
- JSP
- PHP
- DB



필수



선택

- Python
- Spring
- Jquery
- Ajax

# HTTP 프로토콜

## HTTP 프로토콜의 특징

HyperText Transfer Protocol ( 하이퍼 텍스트 전송 프로토콜 )

www에서 쓰이는 핵심 프로토콜로 문서의 전송을 위해 쓰이며,  
오늘날 거의 모든 웹 애플리케이션에서 사용되고 있다.

-> 음성, 화상 등 여러 종류의 데이터를 MIME로 정의하여 전송 가능

HTTP 특징

Request / Response ( 요청/응답 ) 동작에 기반하여 서비스 제공

# HTTP 프로토콜

## HTTP 프로토콜의 특징

### HTTP 1.0의 특징

“연결 수립, 동작, 연결 해제”의 단순함이 특징

-> 하나의 URL은 하나의 TCP 연결

HTML 문서를 전송 받은 뒤 연결을 끊고 다시 연결하여 데이터를 전송한다.

### HTTP 1.0의 문제점

단순 동작 ( 연결 수립, 동작, 연결 해제 )이 반복되어 통신 부하 문제 발생

# HTTP 프로토콜

## HTTP 프로토콜의 특징

HTTP 1.1의 특징

HTTP 1.0과 호환 가능

Multiple Request 처리가 가능하여 Client의 Request가 많을 경우  
연속적인 응답 제공 -> Pipeline 방식의 Request / Response 진행

HTTP 1.0과는 달리 Server가 갖는 하나의 IP Address와  
다수의 Web Site 연결 가능

HTTP 1.1

빠른 속도와 Internet Protocol 설계에 최적화될 수 있도록 Cache 사용  
Data를 압축해서 전달이 가능하도록 하여 전달하는 Data 양이 감소

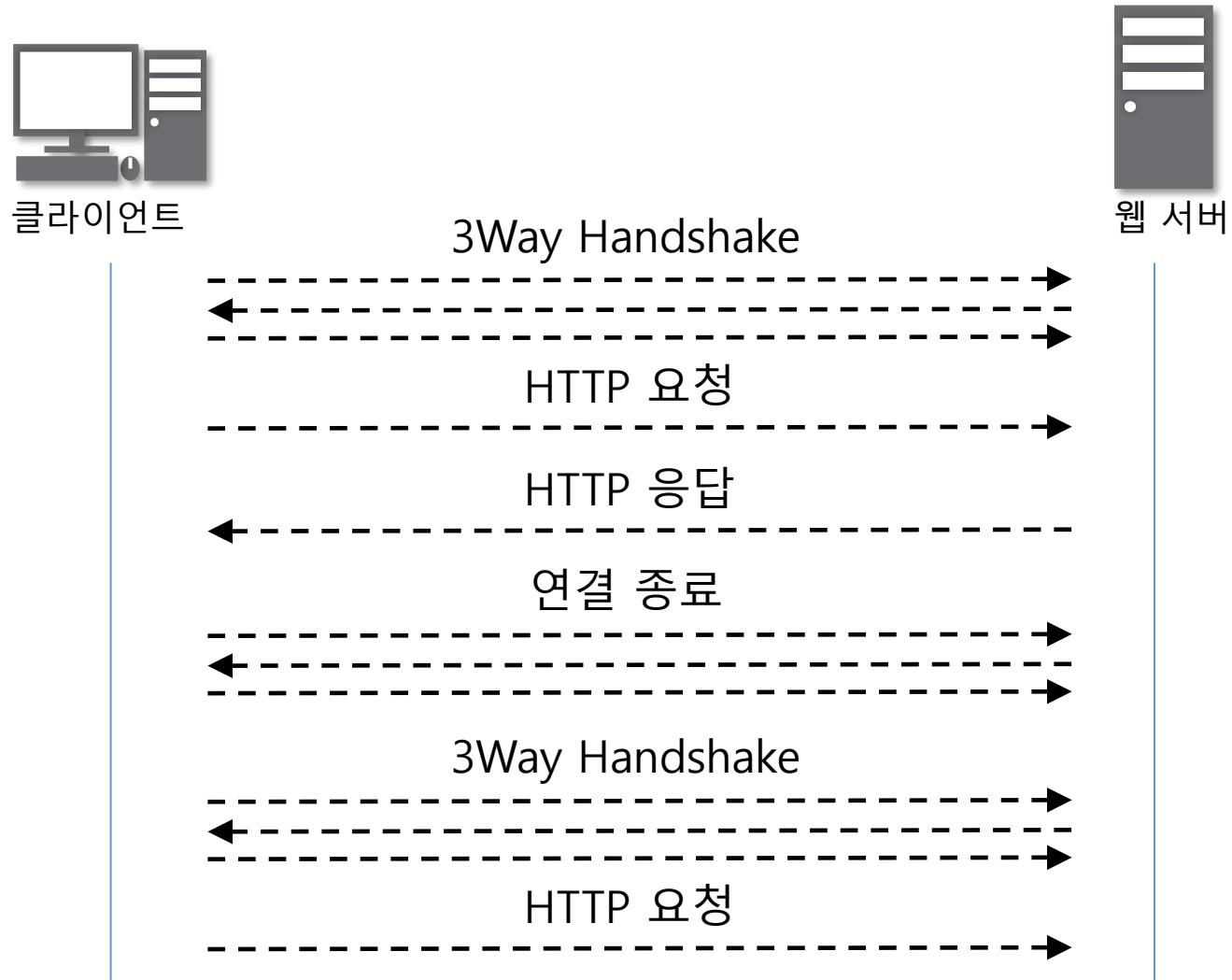
# HTTP 프로토콜

## HTTP 프로토콜의 통신 과정

“

네트워크 부하가 심한  
HTTP/1.0

”



# HTTP 프로토콜

## HTTP 프로토콜의 통신 과정

“

네트워크 부하가 심한  
HTTP/1.0

”



# HTTP 프로토콜

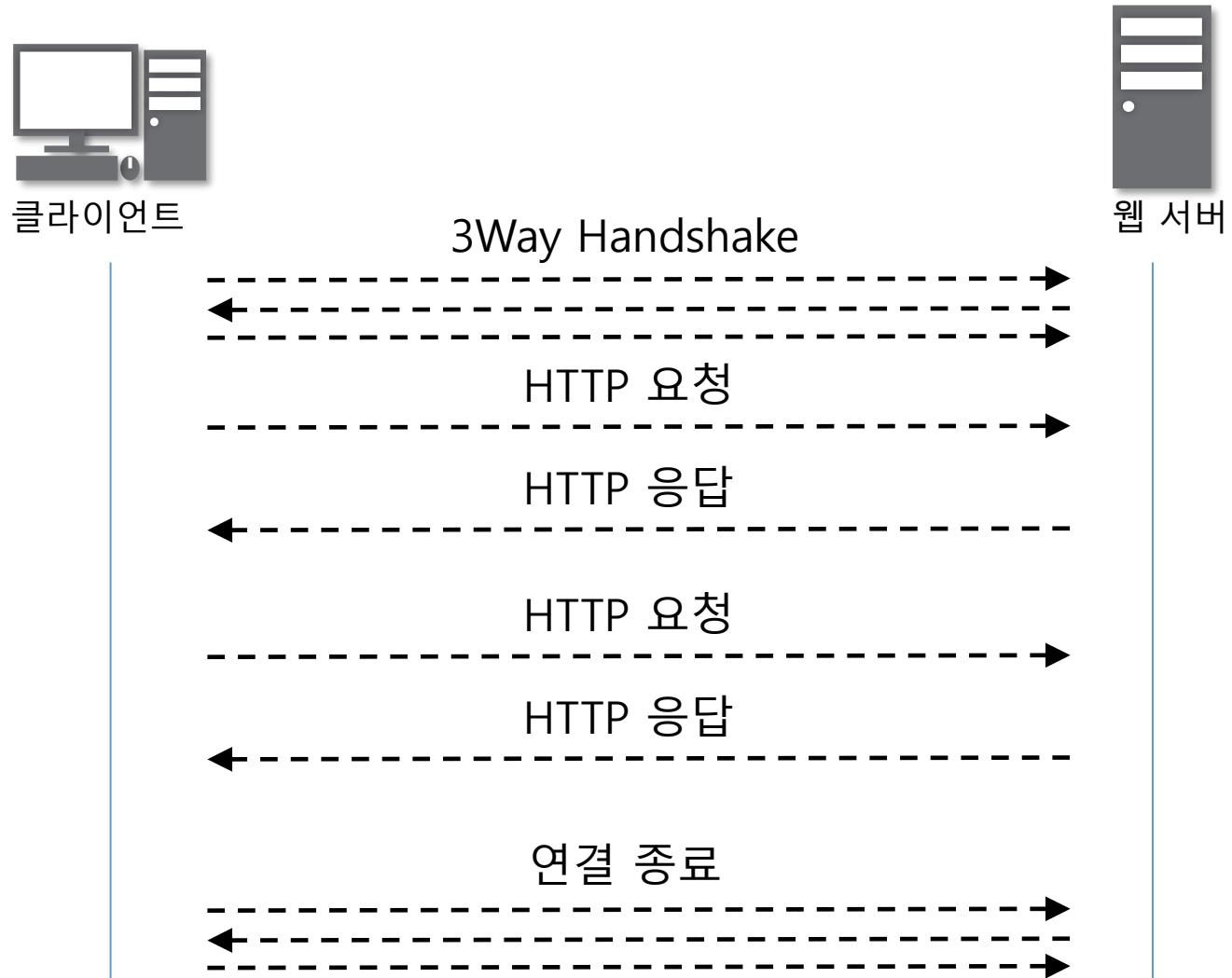
## HTTP 프로토콜의 통신 과정

“

1.0의 문제점을 보완한

HTTP/1.1

”



# HTTP 요청 프로토콜

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

“

요청하는 방식을 정의하고  
**요청 프로토콜 구조**  
클라이언트의 정보를 담고 있는

”



# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

//

요청하는 방식을 정의하고  
**요청 프로토콜 구조**  
클라이언트의 정보를 담고 있는

//

```
GET /produ/content.asp?code=sch-v310 HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, image/pjpeg, application/x-shockwave-flash,
application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword,
application/xaml+xml, application/x-ms-xbap, application/x-ms-application, /*/
Referer: http://www.sst.com/
Accept-Language: ko
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0;
InfoPath.3; .NET4.0C; .NET4.0E)
Accept-Encoding: gzip, deflate
Host: www.sst.com
Proxy-Connection: Keep-Alive
Cookie: ASPSESSIONIDCCDQARAS=EMCDFFBCECFHKPAGOADOIOIE
```

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

〃

요청하는 방식을 정의하고  
**요청 프로토콜 구조**  
클라이언트의 정보를 담고 있는



〃

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

HTTP 메소드  
요청 방식

〃

메소드 종류	설명
GET	Client가 Server로부터 문서를 읽어오려 할 때 사용
HEAD	Client가 문서가 아닌 문서에 대한 특정 정보를 원할 경우 사용
POST	Client가 Server에게 어떤 정보를 전송할 때 사용
PUT	Client가 Server에 특정 자원을 업로드할 때 사용
PATCH	PUT과 비슷함, 기존 파일에서 변경사항만을 포함
COPY	파일을 다른 위치로 복사하기 위해 사용
MOVE	파일을 다른 위치로 이동하기 위해 사용
DELETE	Server에서 문서를 제거하기 위해 사용
LINK	문서에서 다른 위치로의 링크를 생성하기 위해 사용
UNLINK	LINK Method에 의해 생성된 링크를 삭제하기 위해 사용
OPTION	Client가 Server에게 사용 가능한 옵션을 질의하기 위해 사용

〃

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

HTTP 메소드  
요청 방식

〃

메소드 종류	설명
GET	Client가 Server로부터 문서를 읽어오려 할 때 사용
HEAD	Client가 문서가 아닌 문서에 대한 특정 정보를 원할 경우 사용
POST	Client가 Server에게 어떤 정보를 전송할 때 사용
PUT	Client가 Server에 특정 자원을 업로드할 때 사용
PATCH	PUT과 비슷함, 기존 파일에서 변경사항만을 포함
COPY	파일을 다른 위치로 복사하기 위해 사용
MOVE	파일을 다른 위치로 이동하기 위해 사용
DELETE	Server에서 문서를 제거하기 위해 사용
LINK	문서에서 다른 위치로의 링크를 생성하기 위해 사용
UNLINK	LINK Method에 의해 생성된 링크를 삭제하기 위해 사용
OPTION	Client가 Server에게 사용 가능한 옵션을 질의하기 위해 사용

〃

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

//

HTTP 메소드  
GET 요청 방식

//

```
Wireshark - Follow HTTP Stream (tcp.stream eq 119) · 로컬 영역 연결

GET /user/login HTTP/1.1
Host: 54.180.22.166
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64)
(KHTML, like Gecko) Chrome/75.0.3770.100 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml,
        application/webp,image/apng,*/*;q=0.8,application/signed-exchange
Accept-Encoding: gzip, deflate
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: JSESSIONID=5B4F492FC9F27DDA2410934195FBAD54
```

HTTP/1.1 200

Content-Type: text/html; charset=UTF-8

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

//

HTTP 메소드

POST 요청 방식

//

POST /user/loginPost HTTP/1.1

Host: 54.180.22.166

Connection: keep-alive

Content-Length: 26

Accept-Encoding: gzip, deflate

Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7

Cookie: JSESSIONID=5B4F492FC9F27DDA2410934195FBAD54

uid=[REDACTED] &upw=[REDACTED] HTTP/1.1 302

Location: /sboard/list

Content-Length: 0

Date: Thu, 04 Jul 2019 06:19:42 GMT

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

Client가 특정 페이지를 요청하면서 Server로 보내는 데이터

“



GET 방식과 POST 방식의 차이점

**NAVER 만화**

| 웹소설

제목/작가:

홈

웹툰

베스트 도전

도전만화

“

요일별

장르별

작품별

작가별

연

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

“

GET 방식과 POST 방식의 차이점

“

Client가 특정 페이지를 요청하면서 Server로 보내는 데이터

```
Wireshark - Follow HTTP Stream (tcp.stream eq 119) · 로컬 영역 연결

POST /user/loginPost HTTP/1.1
Host: 54.180.22.166
Connection: keep-alive
Content-Length: 26

Accept-Encoding: gzip, deflate
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: JSESSIONID=5B4F492FC9F27DDA2410934195FBAD54

uid=[REDACTED]&upw=[REDACTED] HTTP/1.1 302
Location: /sboard/list
Content-Length: 0
Date: Thu, 04 Jul 2019 06:19:42 GMT
```

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

〃

요청하는 방식을 정의하고  
**요청 프로토콜 구조**  
클라이언트의 정보를 담고 있는



〃

# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

인터넷 상에서 특정 자원(파일)을 나타내는 유일한 주소

“



Uniform Resource Identifier  
URI의 구조

**NAVER 만화**

| 웹소설

제목/작가:

홈

웹툰

베스트 도전

도전만화

“



# HTTP 요청 프로토콜

## HTTP 요청 프로토콜의 구조

---

“

Uniform Resource Identifier  
URI의 구조

scheme ://host[:port][/path][?query]  
ex) ftp ://IP주소 :포트 /파일이름  
http ://IP주소 :포트 /폴더이름/파일이름  
도메인주소

”

---

# HTTP 응답 프로토콜

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

“

사용자가 볼 웹 페이지를 담고 있는  
응답 프로토콜 구조

”



# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

〃

사용자가 볼 웹 페이지를 담고 있는  
응답 프로토콜 구조

〃

```
HTTP/1.1 200 OK
Date: Fri, 25 Mar 2011 06:54:45 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Content-Length: 93639
Content-Type: text/html
Set-Cookie: ASPSESSIONIDACAQARBT=HMJLELBCDNGEJCLNAMJFLCBO; path=/
Cache-control: private
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=euc-kr">
<title>『Magicimt』</title>
<style type="text/css">
...

```

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

“

사용자가 볼 웹 페이지를 담고 있는  
응답 프로토콜 구조



“

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

“

서버가 알려주는 여러가지 정보  
상태 코드

”

상태 코드 종류	설명
100 ~ 199	단순한 정보
200 ~ 299	Client의 요청이 성공
300 ~ 399	Client의 요청이 수행되지 않아 다른 URL로 재지정
400 ~ 499	Client의 요청이 불완전하여 다른 정보가 필요
500 ~ 599	Server의 오류를 만나거나 Client의 요청 수행 불가

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

“

서버가 알려주는 여러가지 정보  
상태 코드

”

상태 코드 종류	설명
100 ~ 199	단순한 정보
200 ~ 299	Client의 요청이 성공
300 ~ 399	Client의 요청이 수행되지 않아 다른 URL로 재지정
400 ~ 499	Client의 요청이 불완전하여 다른 정보가 필요
500 ~ 599	Server의 오류를 만나거나 Client의 요청 수행 불가

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

〃

성공적인 통신  
200 OK

상태 코드 종류	상태 문구	설명
200	OK	Client의 요청이 성공했다는 것을 나타낸다

〃

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

〃

클라이언트의 실수, 잘못, 오류  
400번대

상태 코드 종류	상태 문구	설명
403	Forbidden	Client가 권한이 없는 페이지를 요청했을 때
404	Not Found	Client가 서버에 없는 페이지를 요청했을 때

〃

# HTTP 응답 프로토콜

## HTTP 응답 프로토콜의 구조

〃

서버의 실수, 잘못, 오류  
500번대

상태 코드 종류	상태 문구	설명
500	Internal Server Error	Server의 일부가 멈췄거나 설정 오류가 발생
503	Service Unavailable	최대 Session 수를 초과했을 때

〃

# HTTP 헤더 포맷

# HTTP 헤더 포맷

## HTTP 헤더 구조

//

수많은 정보를 담고 있는  
HTTP 헤더

//



# HTTP 헤더 포맷

## HTTP 헤더 구조

〃

수많은 정보를 담고 있는  
HTTP 헤더

〃

**Request Line**

일반, 요청, 항목 헤더

공백

**Body**

**Status Line**

일반, 응답, 항목 헤더

공백

**Body**

# HTTP 헤더 포맷

## 일반 헤더

---

〃

일반적인 정보를 담고 있는  
일반 헤더

헤더 종류	설명
Content-Length	메시지 바디 길이를 나타낼 때 쓰인다
Content-Type	메시지 바디에 들어있는 컨텐츠 종류 ( Ex: HTML 문서는 text/html )

〃

---

# HTTP 헤더 포맷

## 요청 헤더

“

클라이언트 정보를 담고 있는  
요청 헤더

헤더 종류	설명
Cookie	서버로부터 받은 쿠키를 다시 서버에게 보내주는 역할을 한다
Host	요청된 URL에 나타난 호스트명을 상세하게 표시 (HTTP 1.1은 필수)
User-Agent	Client Program에 대한 식별 가능 정보를 제공

”

# HTTP 헤더 포맷

## 응답 헤더

〃

서버 정보를 담고 있는  
응답 헤더

헤더 종류	설명
Server	사용하고 있는 웹서버의 소프트웨어에 대한 정보를 포함
Set-Cookie	쿠키를 생성하고 브라우저에 보낼 때 사용. 해당 쿠키 값을 브라우저가 서버에게 다시 보낼 때 사용한다

〃

실습

## 1. HTTP 프로토콜 작성 실습

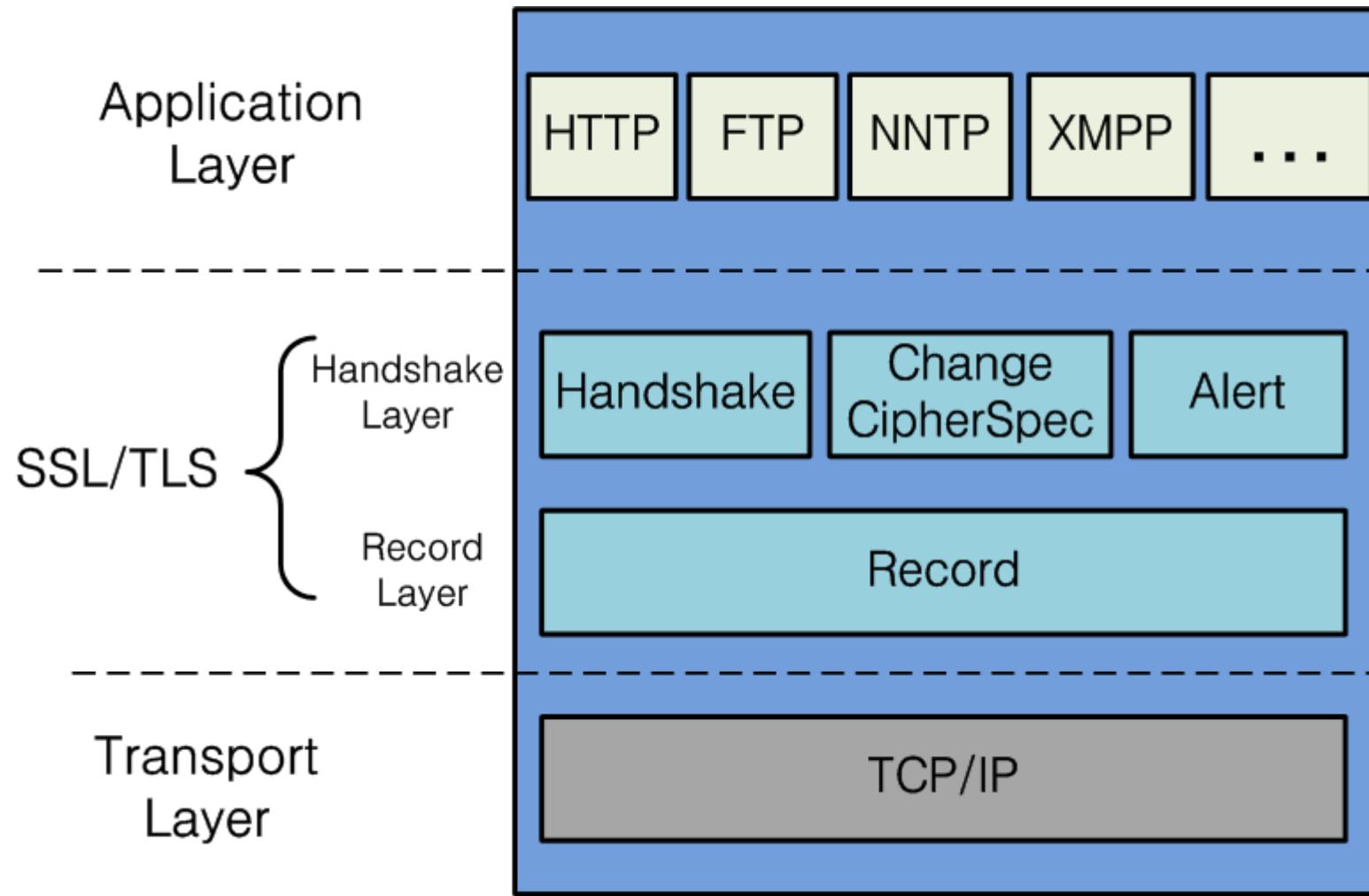
Netcat을 이용하여 HTTP 프로토콜을 직접 작성해보기

## 2. HTTP 프로토콜 수정 실습

HTTP 요청과 응답 프로토콜을 각각 캡쳐해보고 수정해보기

# Burp Suite Community Edition

<https://portswigger.net/burp/communitydownload>



## 1. 자바(JAVA) 설치

01. 먼저 웹 브라우저로 아래의 URL로 접속한다.

- 자바다운로드 페이지: <https://www.oracle.com/downloads/index.html>

02 다운로드 페이지에서 [Java] 선택 → [Java(JDK) for Developers] 선택 [Download] 선택한다(오라클 사이트가 리뉴얼되면 JDK 다운로드 위치와 버전이 변경될 수 있다. 파일럿 프로젝트 [Java SE 8uxxxx]에서 [JDK]에서는 Java SE 8uxxxx 버전이면 문제 없이 진행할 수 있다).

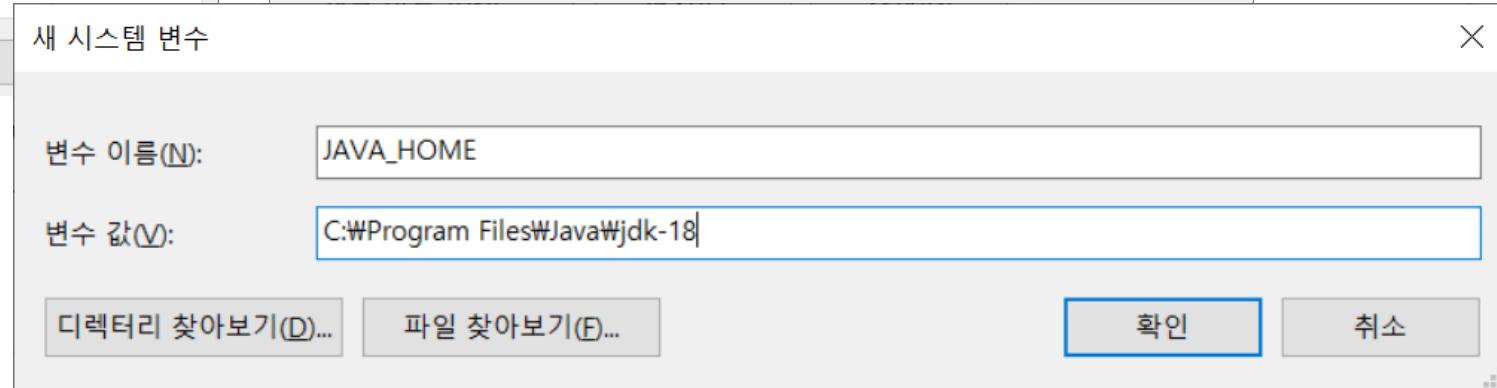
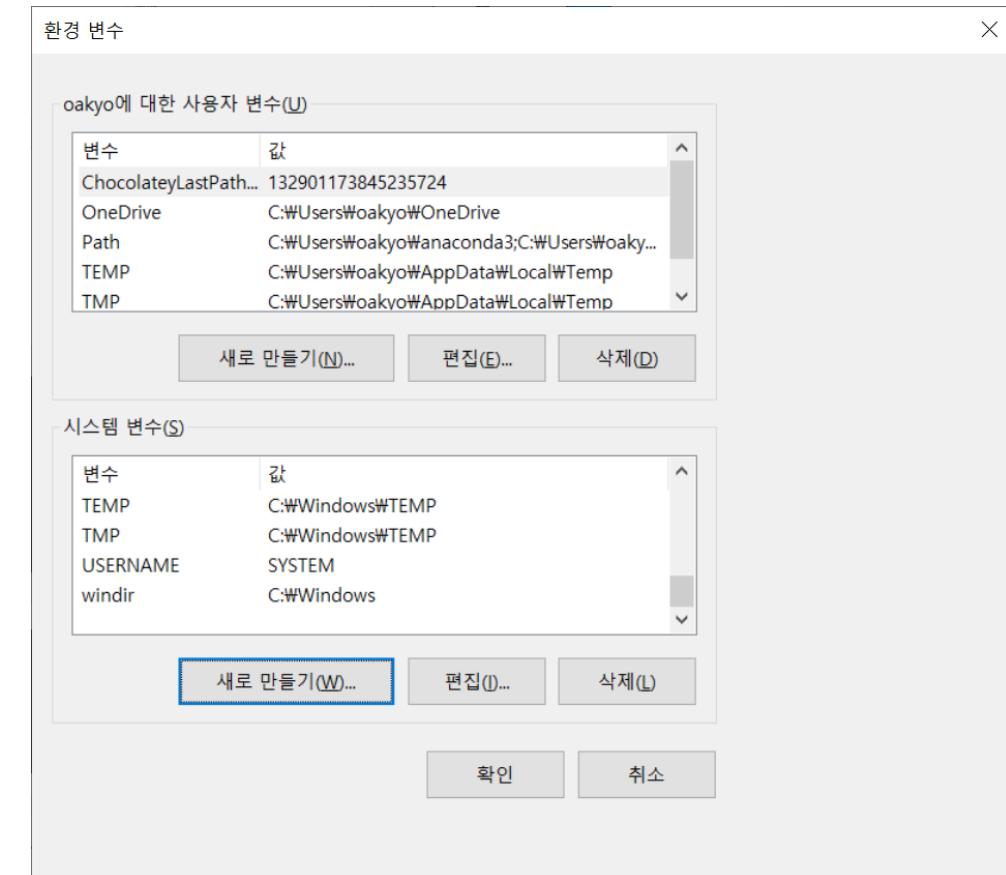
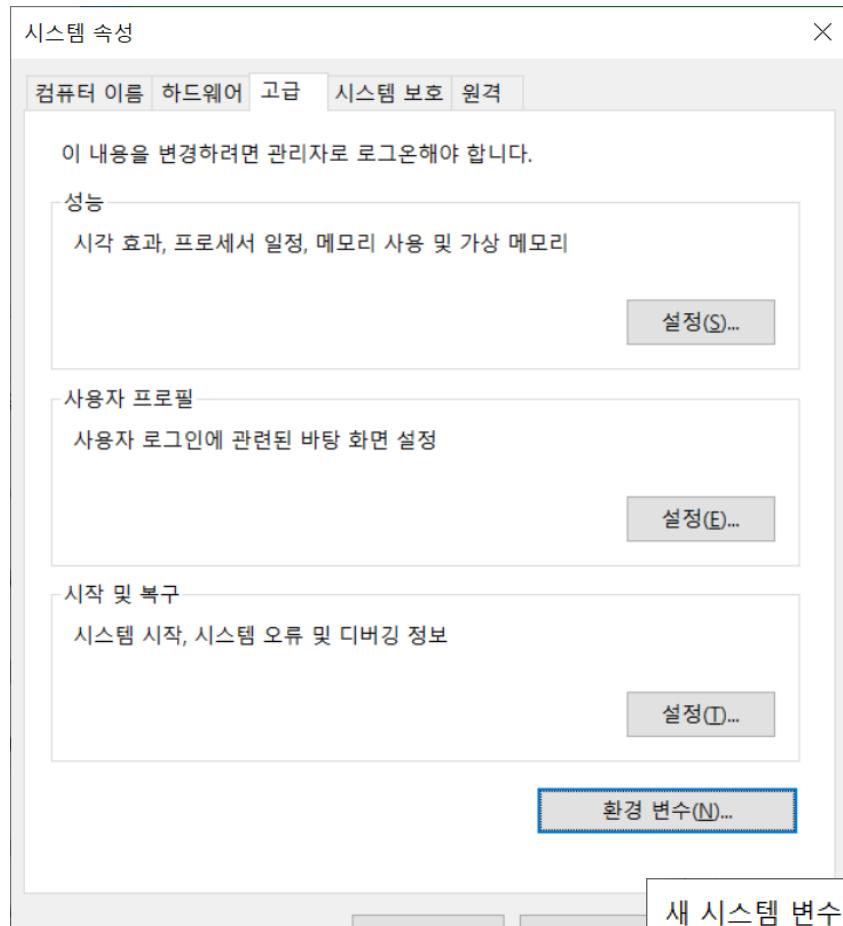
03. 설치가 완료되면 간단히 JAVA\_HOME 환경변수를 설정해 보자(여기서는 윈도우 7 기준으로 설명 한다).

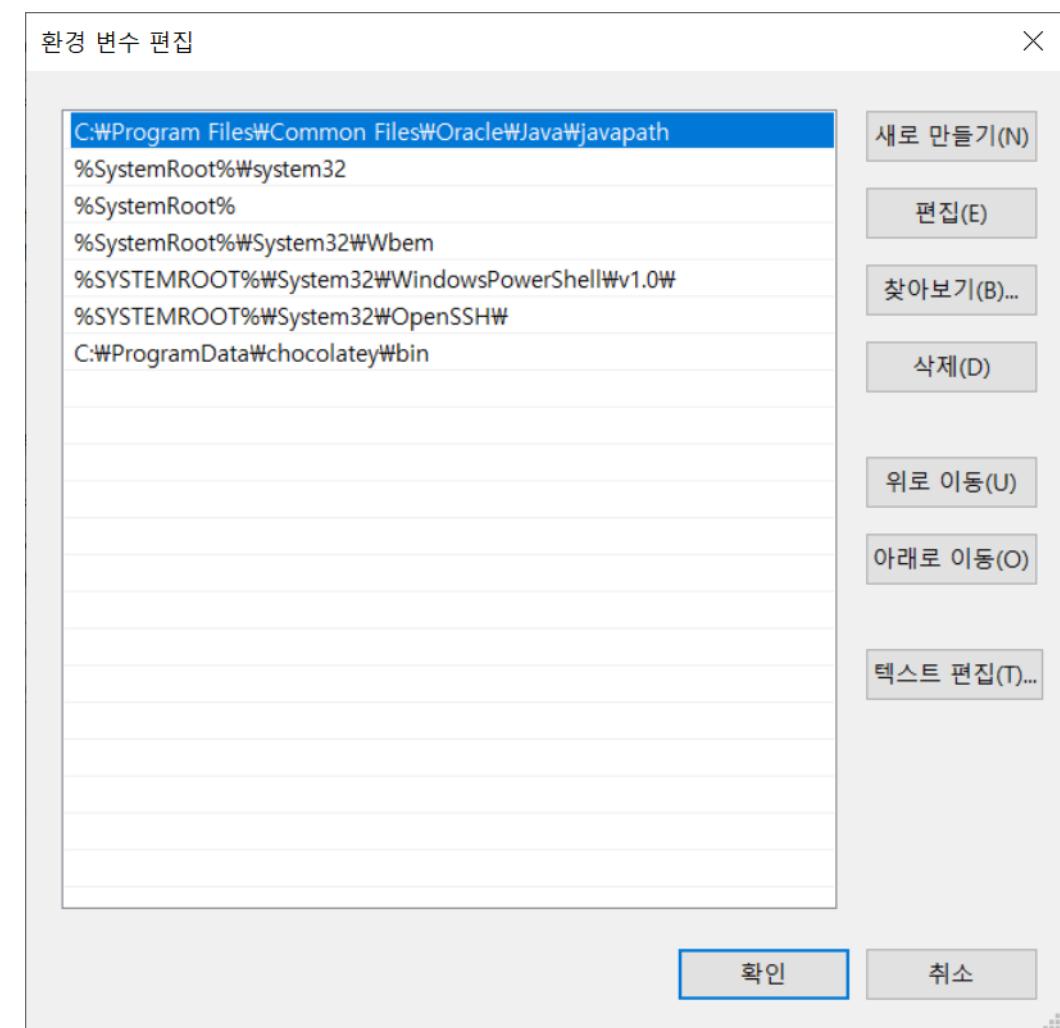
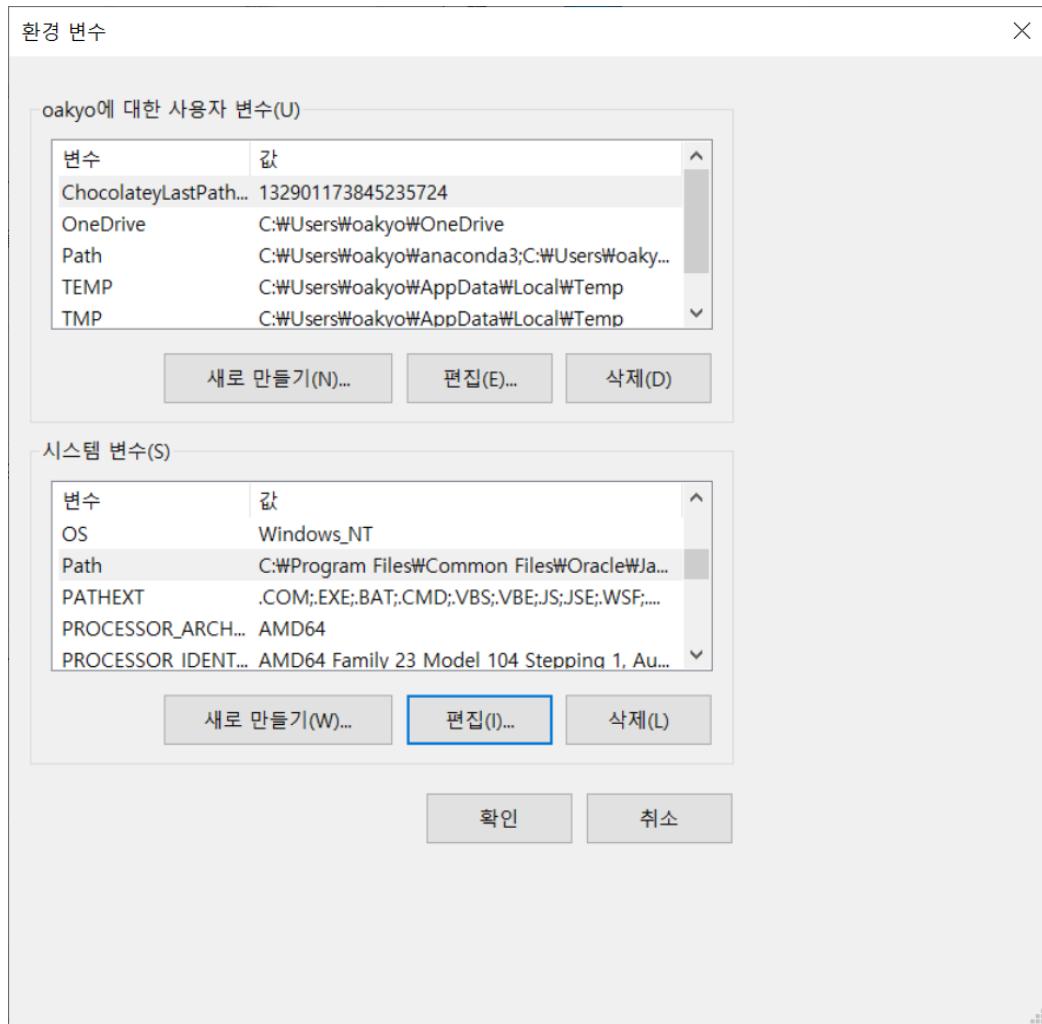
[제어판]-[시스템] → [고급 시스템 설정] [환경변수] 버튼을 차례로 클릭한다.

The screenshot shows a web browser window with the Oracle Java Downloads page open. The URL in the address bar is <https://www.oracle.com/java/technologies/downloads/#jdk18-windows>. The page has a dark header with the Oracle logo and navigation links for Products, Industries, Resources, Customers, Partners, Developers, and Events. Below the header, there are links for Java downloads, Tools and resources, and Java archive. A banner at the top of the main content area states: "Java 18 will receive updates under reserved builds until at least September 2024". The main content is titled "Java SE Development Kit 18 downloads". It includes a brief description of the JDK and its tools. Below this, there are three download options for Windows: "x64 Compressed Archive" (172.54 MB), "x64 Installer" (153.2 MB), and "x64 MSI Installer" (152.08 MB). Each option has a download link and a SHA256 checksum link. A sidebar on the right contains "JDK Script-friendly URLs" information, stating that the URLs listed will remain the same for JDK update releases. At the bottom, there is a file download progress bar for "jdk-18\_windows-x...exe" (108/153MB, 3초 남음).

Product/file description	File size	Download
x64 Compressed Archive	172.54 MB	<a href="https://download.oracle.com/java/18/latest/jdk-18_windows-x64_bin.zip">https://download.oracle.com/java/18/latest/jdk-18_windows-x64_bin.zip</a> (sha256
x64 Installer	153.2 MB	<a href="https://download.oracle.com/java/18/latest/jdk-18_windows-x64_bin.exe">https://download.oracle.com/java/18/latest/jdk-18_windows-x64_bin.exe</a> (sha256
x64 MSI Installer	152.08 MB	<a href="https://download.oracle.com/java/18/latest/jdk-18_windows-x64_bin.msi">https://download.oracle.com/java/18/latest/jdk-18_windows-x64_bin.msi</a> (sha256

<https://www.oracle.com/java/technologies/downloads/#jdk18-windows>





04. JAVA\_HOME 환경변수를 설정한다. [시스템 변수의 새로 만들기]를 클릭해서 아래와 같이 설정한다.

- 변수 이름: JAVA\_HOME
- 변수 값: C:\Program Files\Java\jdk1.8.0\_241

변수 값의 경우 반드시 앞서 설치한 자바의 설치 디렉터리를 지정해야 한다.

- 변수 값에 추가할 내용 : %JAVA\_HOME%\bin

05. 자바 버전 확인

C:> java --version

## 오라클 버추얼 박스 설치

버추얼 박스 역시 설치 방법이 간단하다. 아래의 URL을 통해 설치 VirtualBox 파일을 다운로드한 후 실행하면 안내에 따라 쉽게 설치할 수 있으므로 버추얼 박스의 상세 설치 과정도 생략한다.

오라클 버추얼 박스 다운로드 페이지: <https://www.virtualbox.org/>

(VirtualBox는 하드웨어와 OS의 특성을 많이 타는 소프트웨어다. 파일럿 환경에서 문제가 발생할 경우 많은 해결 사례들을 인터넷 상에서 찾아볼 수 있으니 참고하기 바란다.)

-  일반
-  입력
-  업데이트
-  언어
-  디스플레이
-  네트워크
-  확장
-  프록시

## 네트워크

### NAT 네트워크(N)

활성화됨	이름
<input checked="" type="checkbox"/>	NatNetwork

네트워크 이름: NatNetwork  
네트워크 CIDR: 10.0.2.0/24  
DHCP 지원: 예  
IPv6 지원: 아니요


확인

취소

## 네트워크(N)



만들기(C)



삭제(R)



속성(P)

이름

VirtualBox Host-Only Ethernet Adapter

IPv4 주소/마스크

IPv6 주소/마스크

DHCP 서버

192.168.56.1/24

 사용함

어댑터(A)

DHCP 서버(D)

- 자동으로 어댑터 설정(A)
- 수동으로 어댑터 설정(M)

IPv4 주소(I): 192.168.56.1

IPv4 서브넷 마스크(M): 255.255.255.0

IPv6 주소(P): fe80::e405:f545:8a6d:ec5f

IPv6 접두사 길이(L): 64

초기화

적용

닫기

## 네트워크(N)



만들기(C)



삭제(R)



속성(P)

이름

VirtualBox Host-Only Ethernet Adapter

IPv4 주소/마스크

IPv6 주소/마스크

DHCP 서버

192.168.56.1/24

 사용함

어댑터(A)

DHCP 서버(D)

 서버 사용함(E)

서버 주소(R): 192.168.56.100

서버 마스크(M): 255.255.255.0

최저 주소 한계(L): 192.168.56.101

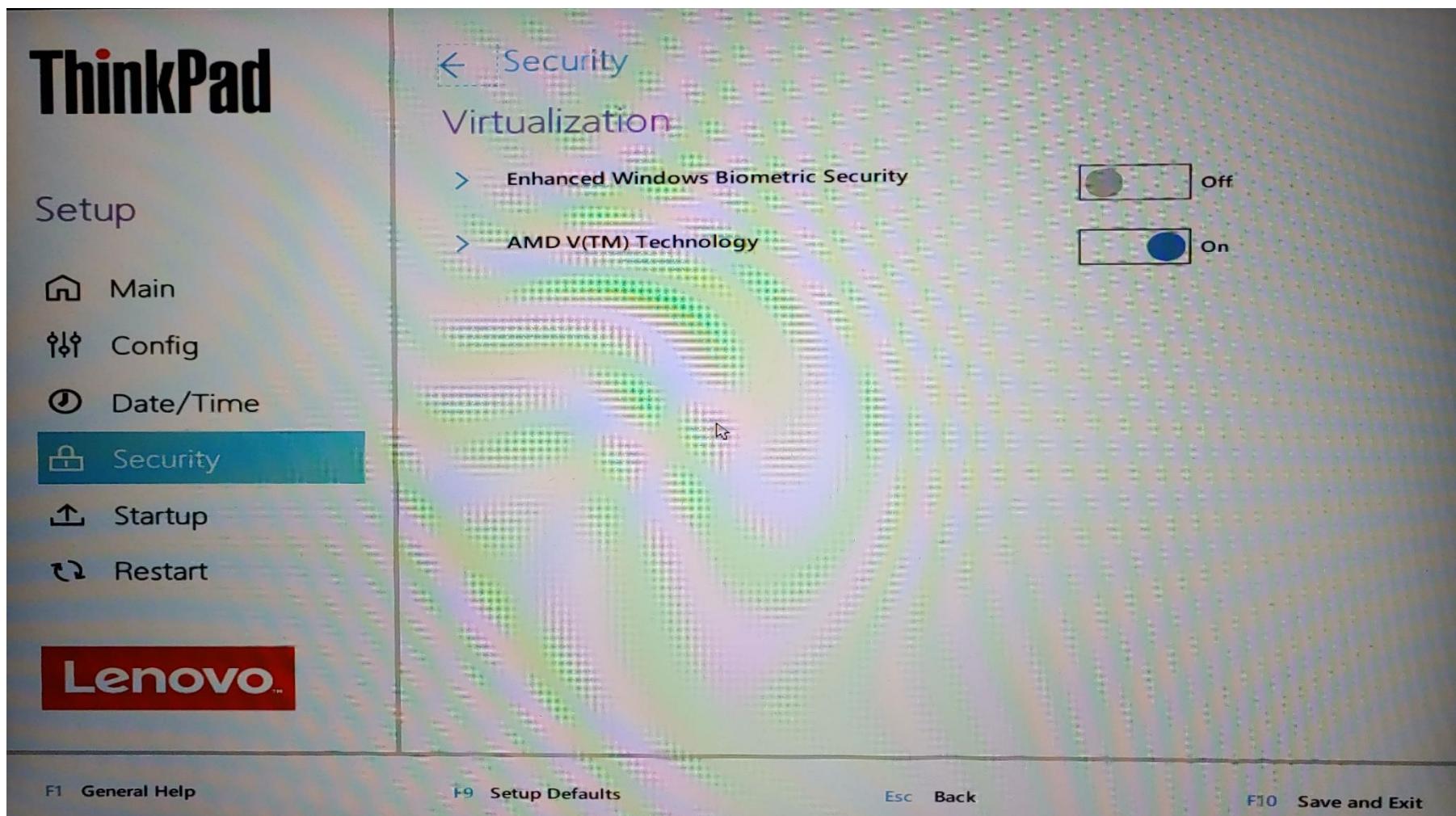
최고 주소 한계(U): 192.168.56.254

초기화

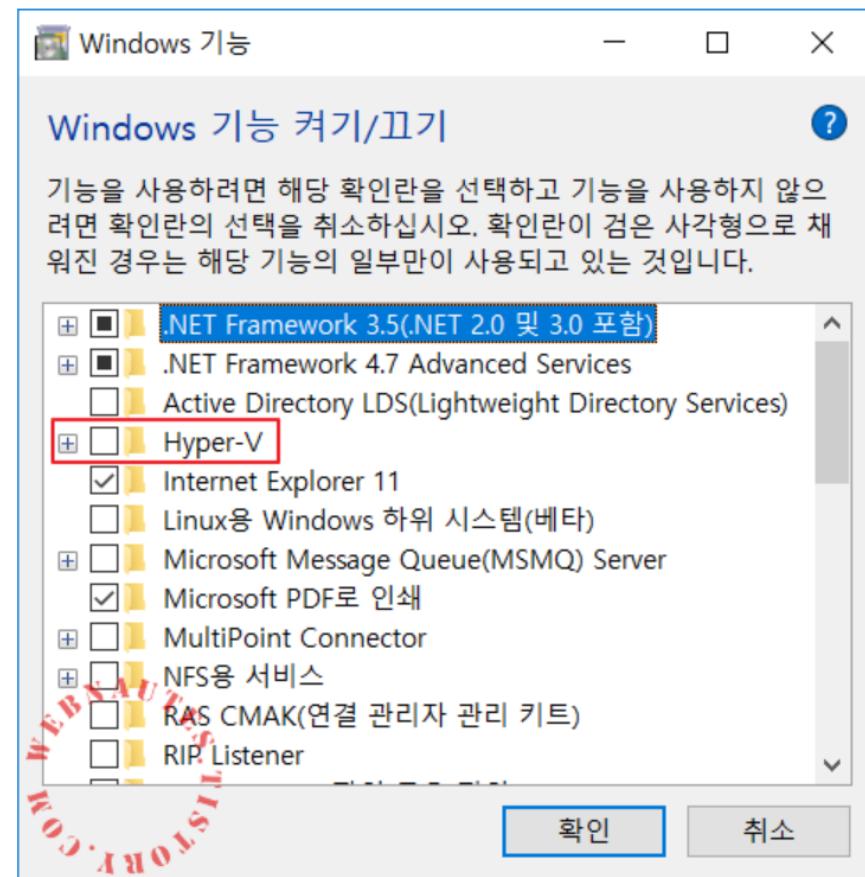
적용

닫기

# PC Bios 가상화 활성 확인



4. Hyper-V 항목을 체크 해제하고 확인을 클릭합니다. 윈도우를 재부팅을 해주어야 적용이 됩니다.



# 리눅스 가상 머신 환경 구성

05. CentOS 7.x 설치 파일을 아래 URL에서 내려받는다. 파일 크기가 1GB에 가까우므로 다운로드가 완료될 때까지 시간이 좀 걸릴 수 있다(파일럿 환경에서는 7.9 버전을 사용한다)

- CentOS 7.x 다운로드 페이지: <https://www.centos.org/centos-linux/>

Debian <http://ftp.harukasan.org/debian-cd/11.3.0-live/i386/iso-hybrid/>  
080027F1B6C1

Mac id: 080027D25D25

-  일반
-  입력
-  업데이트
-  언어
-  디스플레이
-  네트워크
-  확장
-  프록시

## 일반

기본 머신 폴더(M):  C:\WU...akyow\VirtualBox VMs

VRDP 인증 라이브러리(R):  VBoxAuth

확인

취소

파일(F) 머신(M) 네트워크(N) 도움말(H)



도구



만들기(C)



삭제(R)



속성(P)

이름

VirtualBox Host-Only Ethernet Adapter

IPv4 주소/마스크

192.168.28.1/24

IPv6 주소/마스크

DHCP 서버

 사용함

어댑터(A)

DHCP 서버(D)

- 자동으로 어댑터 설정(A)
- 수동으로 어댑터 설정(M)

IPv4 주소(I): IPv4 서브넷 마스크(M): IPv6 주소(P): IPv6 접두사 길이(L): 

적용

초기화

파일(F) 머신(M) 네트워크(N) 도움말(H)



도구



만들기(C)



삭제(R)



속성(P)

이름	IPv4 주소/마스크	IPv6 주소/마스크	DHCP 서버
VirtualBox Host-Only Ethernet Adapter	192.168.28.1/24		<input type="checkbox"/> 사용함

어댑터(A)

DHCP 서버(D)

 서버 사용함(E)

서버 주소(R): 192.168.28.2

서버 마스크(M): 255.255.255.0

최저 주소 한계(L): 192.168.28.3

최고 주소 한계(U): 192.168.28.254

적용

초기화



가상 머신 만들기

## 이름 및 운영 체제

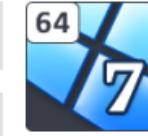
새 가상 머신을 나타내는 이름과 저장할 대상 폴더를 입력하고 설치할 운영 체제를 선택하십시오. 입력한 이름은 VirtualBox에서 가상 머신을 식별하는 데 사용됩니다.

이름:

머신 폴더:

 C:\Users\Woakyoo\VirtualBox VMs

종류(I): Microsoft Windows

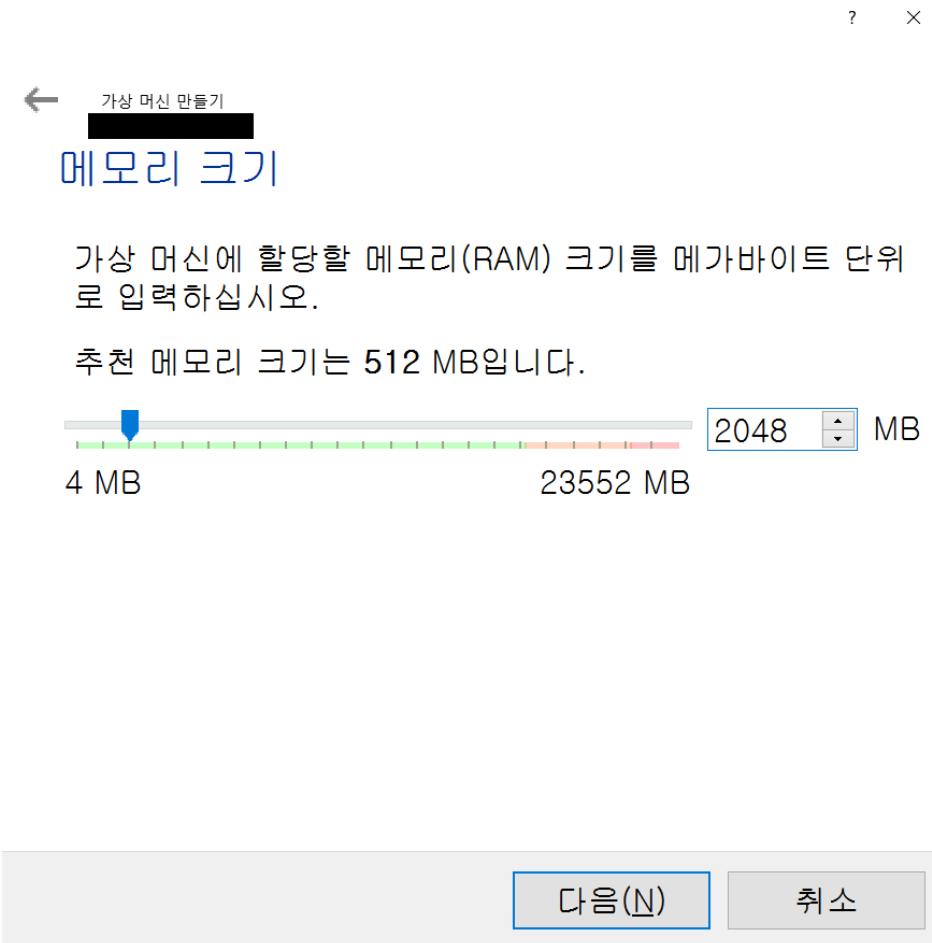
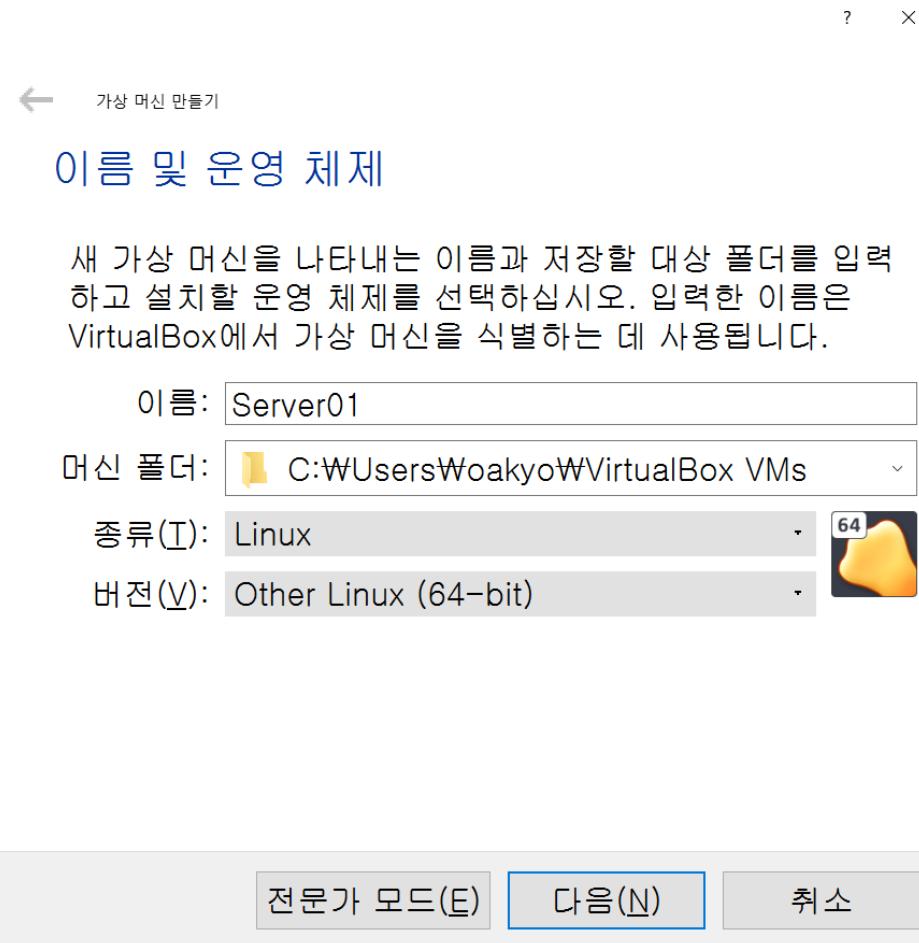


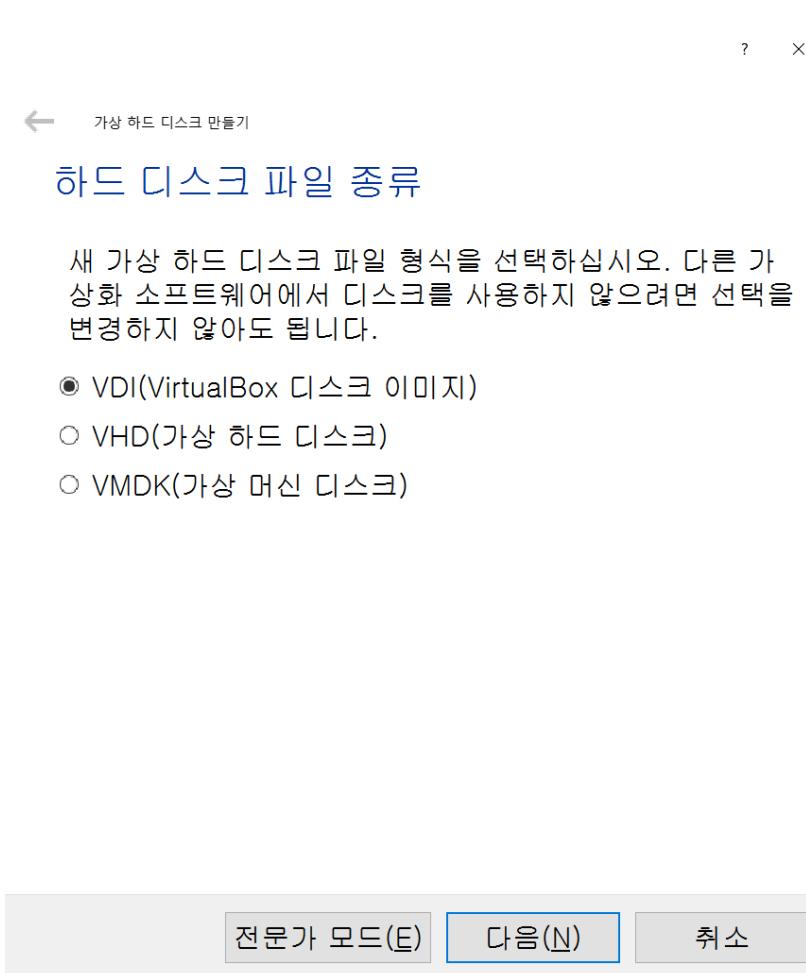
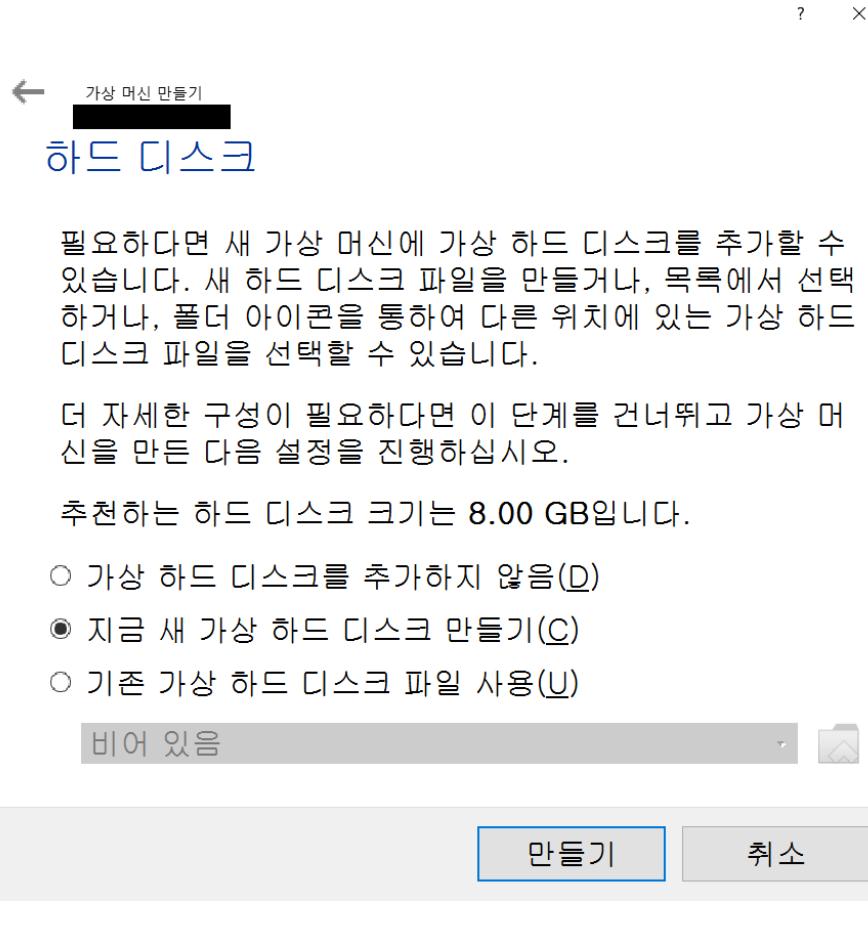
버전(V): Windows 7 (64-bit)

전문가 모드(E)

다음(N)

취소





?    X

← 가상 하드 디스크 만들기

## 물리적 하드 드라이브에 저장

새 가상 하드 디스크 파일을 사용하는 대로 커지게 할 것인지(동적 할당) 최대 크기로 만들 것인지(정적 할당) 선택하십시오.

동적 할당 하드 디스크 파일은 가상 디스크를 사용할 때 고정된 최대 크기까지 파일 크기가 커지지만, 사용량이 줄어들어도 자동적으로 작아지지는 않습니다.

고정 크기 하드 디스크 파일은 만드는 데 더 오래 걸리지만 사용할 때 더 빠릅니다.

동적 할당(D)

고정 크기(F)

다음(N)    취소

← 가상 하드 디스크 만들기

## 파일 위치 및 크기

새 가상 하드 디스크 파일의 이름을 아래 상자에 입력하거나 폴더 아이콘을 클릭해서 파일을 생성할 폴더를 지정할 수 있습니다.

s:\oakyo\VirtualBox VMs\Server01\Server01.vdi

새 가상 하드 디스크 크기를 메가바이트 단위로 입력하십시오. 가상 머신에서 가상 하드 드라이브에 저장할 수 있는 데이터의 최대 크기입니다.



4.00 MB    2.00 TB    30.00 GB

만들기    취소

파일(F) 머신(M) 도움말(H)



도구



Server01



전원 꺼짐



새로 만들기(N)



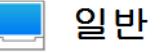
설정(S)



삭제

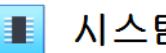


시작(T)

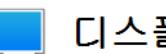


일반

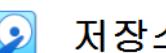
이름: Server01  
운영 체제: Other Linux (64-bit)



기본 메모리: 2048 MB  
부팅 순서: 플로피, 광 디스크, 하드 디스크  
가속: VT-X/AMD-V, 네스티드 페이징, PAE/NX, KVM 반가상화



비디오 메모리: 16 MB  
그래픽 컨트롤러: VMSVGA  
원격 데스크톱 서버: 사용 안 함  
녹화: 사용 안 함



컨트롤러: IDE

IDE 프라이머리 마스터: Server01.vdi (일반, 30.00 GB)

IDE 세컨더리 마스터: [광학 드라이브] 비어 있음



미리 보기



## 파일 머신 보기 입력 장치 도움말

## NETWORK &amp; HOST NAME

Done

## CENTOS 7 INSTALLATION

us

Help! (F1)

- Ethernet (enp0s3)  
Intel Corporation 82540EM Gigabit Ethernet Controller (VirtualBox Host-Only Adapter)
- Ethernet (enp0s8)  
Intel Corporation 82540EM Gigabit Ethernet Controller (VirtualBox Host-Only Adapter)

Ethernet (enp0s3)  
Disconnected

Hardware Address 08:00:27:B8:9E:06

Speed 1000 Mb/s



Configure...

Host name: server01.hadoop.com

Apply

Current host name: localhost



Right Control

## 파일 머신 보기 입력 장치 도움말

## NETWORK &amp; HOST NAME

Done

## CENTOS 7 INSTALLATION

us

Help! (F1)

- Ethernet (enp0s3)  
Intel Corporation 82540EM Gigabit Ethernet Controller (VirtualBox Host-Only Adapter)
- Ethernet (enp0s8)  
Intel Corporation 82540EM Gigabit Ethernet Controller (VirtualBox Host-Only Adapter)

Ethernet (enp0s3)  
Disconnected

Hardware Address 08:00:27:B8:9E:06

Speed 1000 Mb/s



Configure...

Host name: server01.hadoop.com

Apply

Current host name: localhost



Right Control

-  일반
-  시스템
-  디스플레이
-  저장소
-  오디오
-  네트워크
-  직렬 포트
-  USB
-  공유 폴더
-  사용자 인터페이스

## 네트워크

어댑터 1 어댑터 2 어댑터 3 어댑터 4

네트워크 어댑터 사용하기(E)

다음에 연결됨(A): NAT

이름(N):

▼ 고급(D)

어댑터 종류(I): Intel PRO/1000 MT Desktop(82540EM)

무작위 모드(P): 거부

MAC 주소(M): 08002780A1FC

케이블 연결됨(C)

포트 포워딩(P)

확인 취소

-  일반
-  시스템
-  디스플레이
-  저장소
-  오디오
-  네트워크
-  직렬 포트
-  USB
-  공유 폴더
-  사용자 인터페이스

## 네트워크

어댑터 1 어댑터 2 어댑터 3 어댑터 4

네트워크 어댑터 사용하기(E)

다음에 연결됨(A): 호스트 전용 어댑터

이름(N): VirtualBox Host-Only Ethernet Adapter

▼ 고급(D)

어댑터 종류(I): Intel PRO/1000 MT Desktop(82540EM)

무작위 모드(P): 모두 허용

MAC 주소(M): 080027139F35

케이블 연결됨(C)

포트 포워딩(P)

확인 취소

- 일반
- 시스템
- 디스플레이
- 저장소
- 오디오
- 네트워크
- 직렬 포트
- USB
- 공유 폴더
- 사용자 인터페이스

## 저장소

저장 장치(S)

- 컨트롤러: IDE
  - Server01.vdi
  - CentOS-7-x86\_64-Minimal-...

속성

광학 드라이브(D): IDE 세컨더리 마스터  라이브 CD/DVD(L)

정보

종류: 이미지  
크기: 973.00 MB  
위치: C:\Users\oakyoo\Desktop\CentOS-7-x86\_64-Minimal-15.10\CentOS-7-x86\_64-Minimal-15.10.vdi

다음에 연결됨: --

**확인** **취소**