

Chapter 9

User Datagram Protocol (UDP)

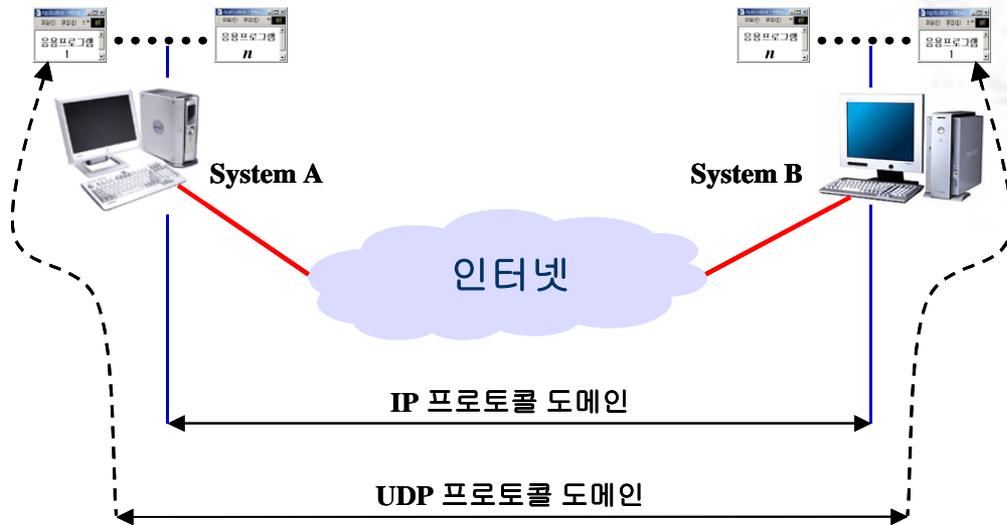


■ OBJECTIVES:

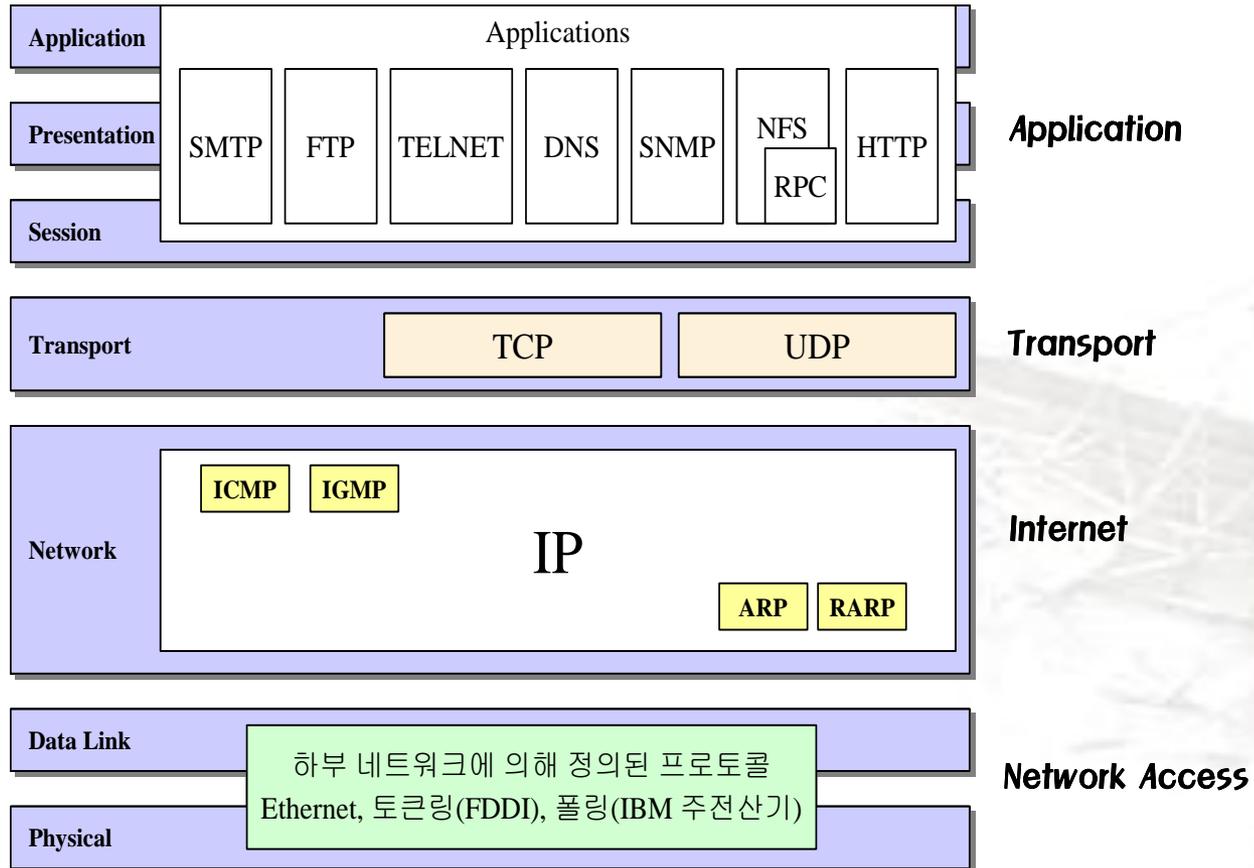
- UDP의 소개 및 TCP/IP 프로토콜 모음의 다른 프로토콜과의 관계를 설명한다.
- 사용자 데이터그램이라고 하는 UDP 패킷의 형식과 헤더 내의 각 필드의 사용법을 설명한다.
- 프로세스 대 프로세스 통신, 초보적인 오류 제어, 다중화/역다중화, UDP에서 제공하는 서비스를 설명한다.
- 옵션인 검사합을 계산하는 방법과 검사합을 계산하는 경우에 송신자가 패킷에 의사 헤더(pseudoheader)를 추가하는 이유에 대해 논의한다.
- 응용 프로그램이 UDP와 같은 단순한 프로토콜로부터 얻을 수 있는 장점을 설명한다.

9-1 개요

UDP는 TCP와 같이 응용 계층과 IP 계층 사이에 위치하며 응용 프로그램과 네트워크 동작 사이의 중개자 역할을 한다. 즉, 인터넷에서 서비스를 제공하는 경우는 TCP나 UDP 프로토콜을 활용한다.



TCP/IP 프로토콜 그룹에서 UDP의 위치



■ UDP 전송 프로토콜 임무

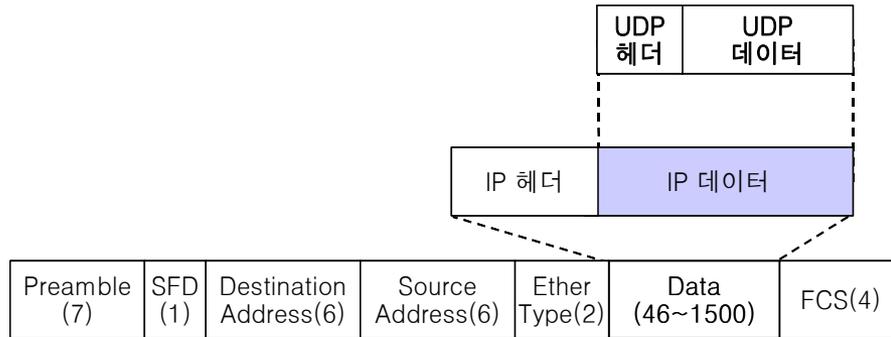
- 프로세스-대-프로세스 통신 생성 : 포트번호 이용
- 최소한의 오류 제어 메커니즘 수행
- 프로세스로부터 데이터 단위를 받아 비연결형, 신뢰성 없는 정보 전달 제공
 - 사용자 데이터 그램이라 하는데, 형식이 매우 단순,
 - 신뢰성 없는 정보 전달 (흐름제어 기능없음): 윈도우 크기, 응답번호, 순서번호가 없음
 - SYN, FIN, ACK등의 데이터 그램 구별 필요없음
- 최소한의 오버헤드만 사용하는 간단한 프로토콜

8-2 USER DATAGRAM

사용자 데이터그램(user datagram)이라는 UDP 패킷은 8바이트의 고정 크기 헤더를 가진다.



사용자 데이터그램 형식



0										1										2										3	
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Source Port(2)										Destination Port(2)																					
Total length(2)										Checksum(2)																					

사용자 데이터그램

■ 발신지 포트 번호(source port number)

➡ 발신지 호스트에서 수행되는 프로세스가 사용하는 포트 번호

■ 목적지 포트 번호(destination port number)

➡ 목적지 호스트에서 수행되는 프로세스가 사용하는 포트 번호

■ 길이(length) : 헤더와 데이터를 합한 전체 길이

■ 검사합(checksum) : 오류 탐지에 사용

Example 9.1

다음과 같이 16진수의 형식으로 UDP 헤더를 덤프한 것이다.

```
04 e1 00 35 00 27 92 27
```

- a. 발신지 포트 번호는 얼마인가?
- b. 목적지 포트 번호는 얼마인가?
- c. 사용자 데이터그램의 총 길이는 얼마인가?
- d. 데이터의 길이는 얼마인가?
- e. 데이터의 전송 방향이 클라이언트에서 서버쪽으로 인가 아니면 반대의 방향인가?
- f. 클라이언트 프로세스는 무엇인가?

Example 9.2

Solution

- a. 발신지 포트 번호는 처음 네 자리 16진수($04E1_{16}$)이며, 10진수로 표현하면 1,249이다.
- b. 목적지 포트 번호는 두 번째 네 자리 16진수(0035_{16})이며, 10진수로 표현하면 53이다.
- c. 세 번째 네 자리 16진수(0027_{16})은 UDP 패킷의 전체 길이를 나타내며, 39바이트이다.
- d. 데이터의 길이는 패킷의 전체 길이에서 헤더의 길이를 뺀 것으로, $39 - 8 = 31$ 바이트이다.
- e. 목적지 포트 번호가 53(잘 알려진 포트: DNS)이므로, 클라이언트로부터 서버로 패킷이 전송되었다.
- f. 클라이언트 프로세스는 DNS이다.

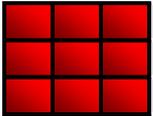
9-3 UDP 서비스

전송 계층 프로토콜에서 제공되는 일반적인 서비스 중 어떤 서비스가 UDP에 의해서 제공되는지 살펴보도록 하자.



UDP 서비스

- 프로세스 대 프로세스 통신 : IP 주소와 포트번호로 구성된 소켓 (주소)이용
- 비연결형 서비스: 각 사용자 데이터그램은 독립적
- 흐름 제어: 기능이 없고 윈도우 메커니즘도 없음
- 오류 제어: 검사합을 제외한 기능 없음
- 혼잡 제어: 기능 없음



UDP에서 사용되는 잘 알려진 포트 번호

포트	서비스	서비스 설명
7	Echo	수신된 사용자 데이터그램을 송신자에게 되돌려 보냄
9	Discard	수신된 사용자 데이터그램을 무시
11	Users	Active users
13	Daytime	날짜와 시간값을 반환
17	Quote	"quote of the day"를 포함하는 문자열을 반환
19	Chargen	임의 길이의 문자열을 반환
53	Nameserver	DNS 이름 서버 프로세스
67	Boothps	설정정보를 다운로드 하는데 사용되는 서버 포트
68	Boothpc	설정정보를 수신하는데 사용되는 클라이언트 포트
69	TFTP	TFTP 서버 프로세스
111	RPC	Remote Procedure Call 서비스를 구현하는데 사용
123	NTP	NTP를 구현하는데 사용
161	SNMP	SNMP 네트워크 관리 질의를 수신하는데 사용
162	SNMP	SNMP 문제점 보고를 받기 위해 사용

검사합 계산을 위한 Pseudoheader

Pseudoheader

32-bit source IP address

32-bit destination IP address

All 0s

8-bit protocol

16-bit UDP total length

Header

Source port address
16 bits

Destination port address
16 bits

UDP total length
16 bits

Checksum
16 bits

Data

(Padding must be added to make the data a multiple of 16 bits)

Example 8.2

7바이트만을 가지고 있는 매우 작은 사용자 데이터그램의 검사합 계산을 보여준다. 데이터의 바이트 수가 홀수이므로 검사합 계산을 위하여 패딩이 필요하다. 사용자 데이터그램이 IP에 전달될 때 패딩과 인사 헤더는 제거될 것이다.



간단한 사용자 데이터그램을 위한 검사합 계산

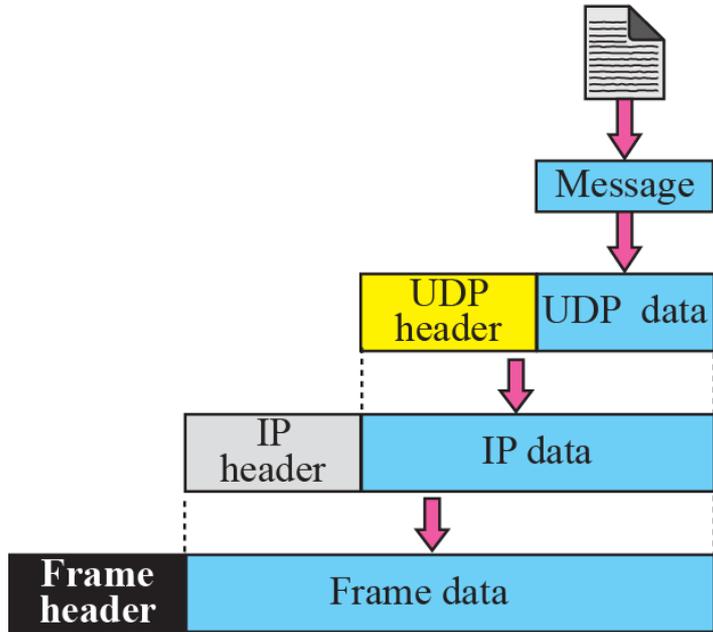
153.18.8.105			
171.2.14.10			
All 0s	17	15	
1087		13	
15		All 0s	
T	E	S	T
I	N	G	Pad

10011001 00010010 → 153.18
 00001000 01101001 → 8.105
 10101011 00000010 → 171.2
 00001110 00001010 → 14.10
 00000000 00010001 → 0 and 17
 00000000 00001111 → 15
 00000100 00111111 → 1087
 00000000 00001101 → 13
 00000000 00001111 → 15
 00000000 00000000 → 0 (checksum)
 01010100 01000101 → T and E
 01010011 01010100 → S and T
 01001001 01001110 → I and N
 01000111 00000000 → G and 0 (padding)

10010110 11101011 → Sum
01101001 00010100 → Checksum

캡슐화(Encapsulation) 역캡슐화 (decapsulation)

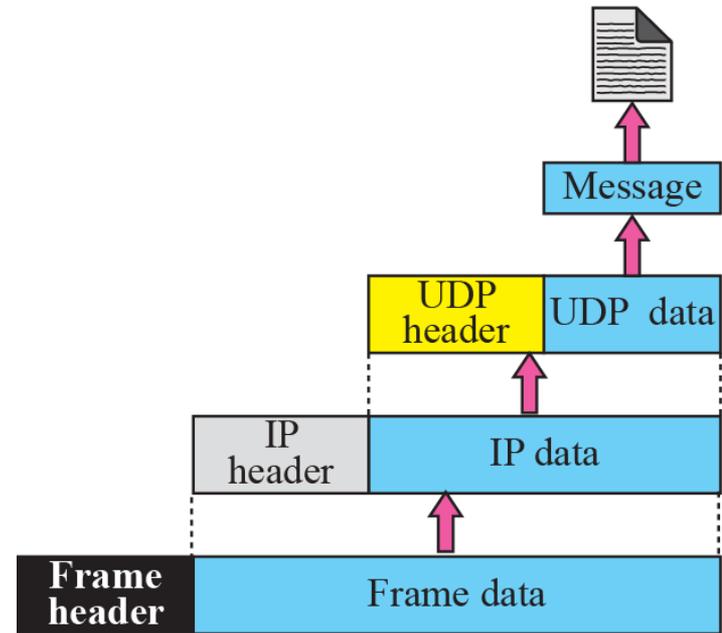
Sender Process



a. Encapsulation

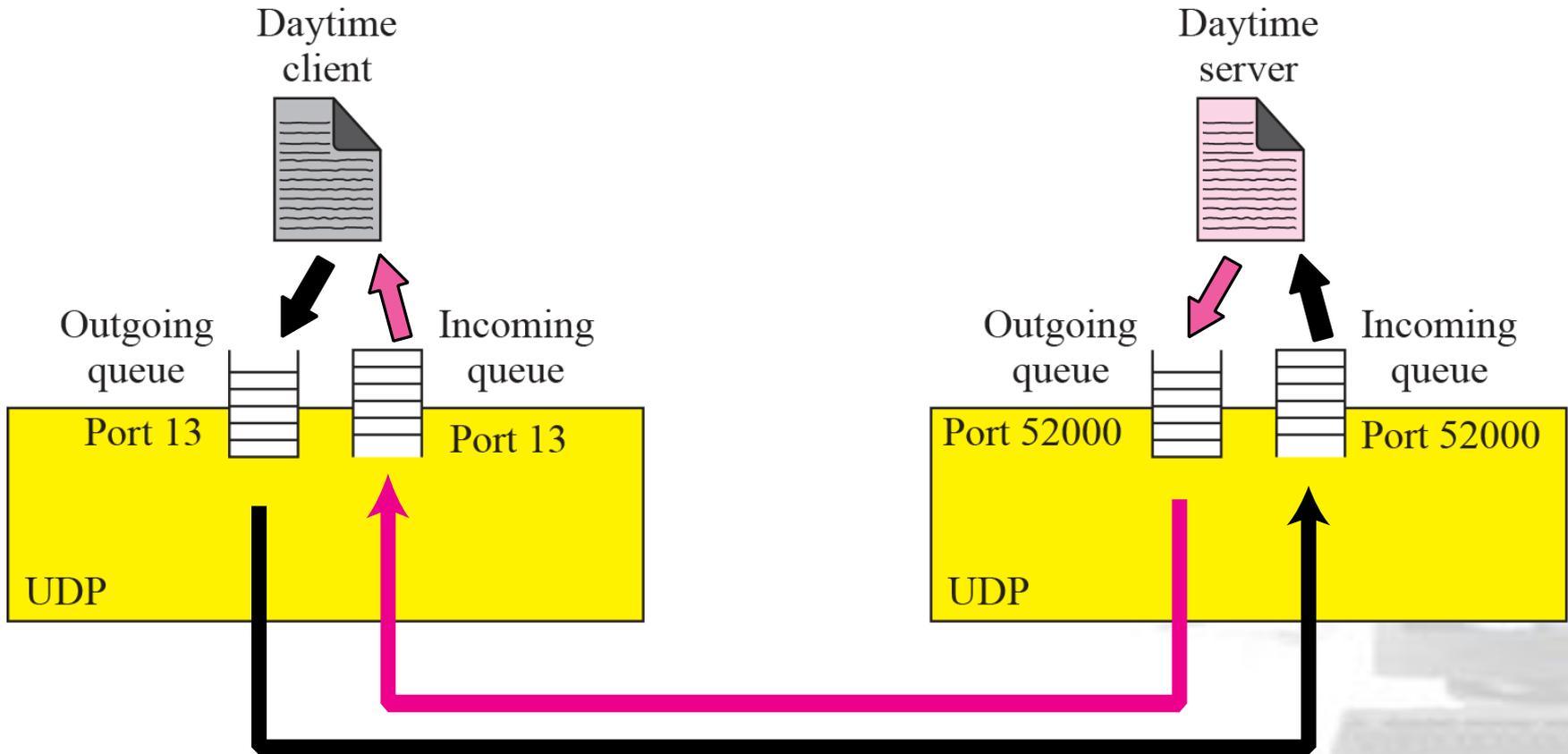


Receiver Process

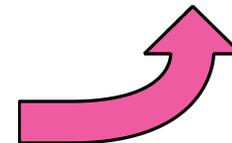
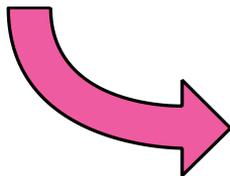
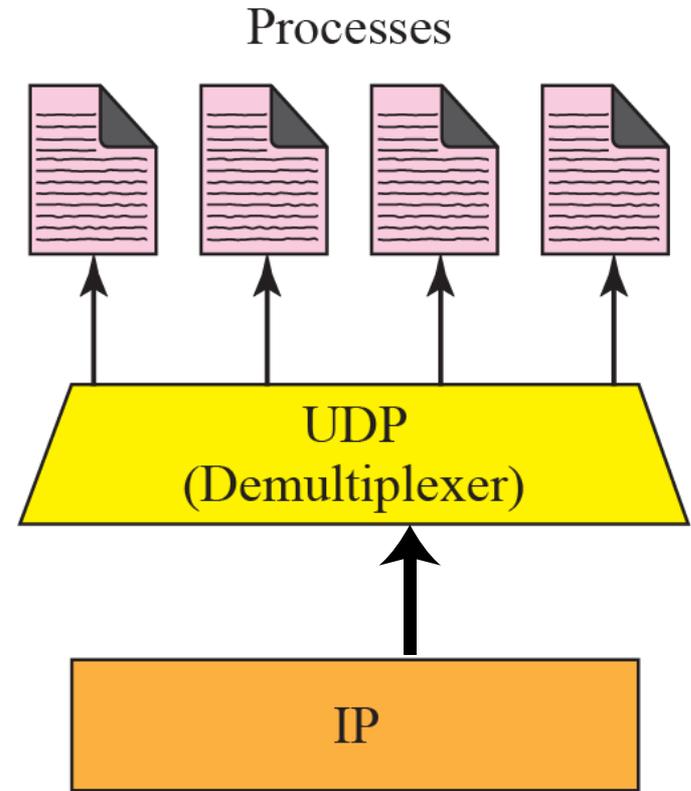
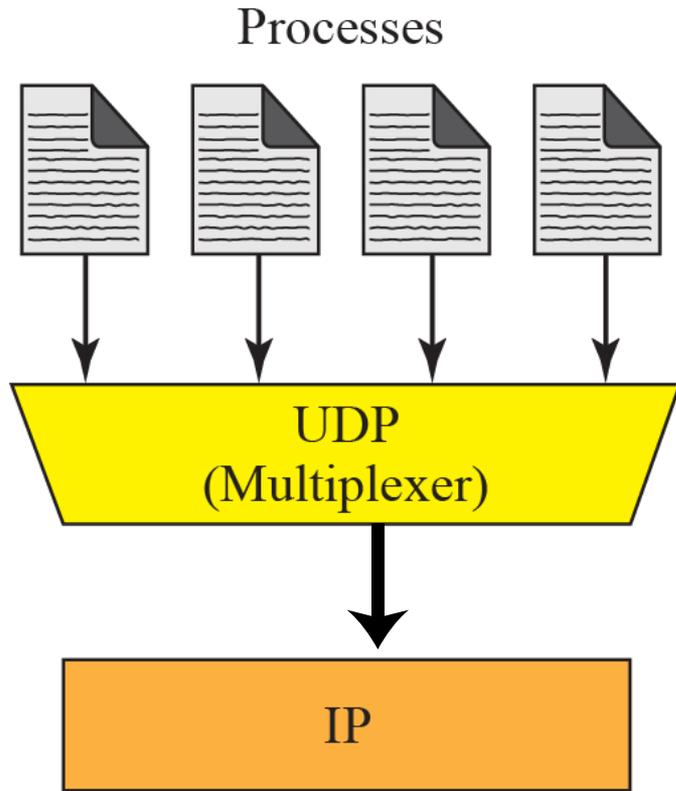


b. Decapsulation

UDP에서 큐(Queue)



다중화(Multiplexing)와 역다중화(demultiplexing)



9-4 UDP 응용

비록 UDP가 신뢰성있는 전송 계층 프로토콜의 어떠한 기준도 만족하지 않지만, UDP를 선호하는 응용들도 있다. 이것에 대한 이유는 어떤 서비스들은 신뢰성 서비스를 용납하지 않거나 선호하지 않는다는 것이다. 응용 설계자는 때때로 칩적을 얻기 위하여 타협해야 하는 경우가 있다.

- DNS 등 간단하고 신뢰성이 크게 요구되지 않고 빠른 정보 전달

(TCP 사용하면

- . 접속시 3개 패킷 : SYN, SYN+ACK, ACK
- . 해제시 2-3개 패킷 소모)

종합문제: 이 문제는 반드시 풀고 갈것

이더넷 상에서 다음과 같이 16진수의 형식의 인터넷 정보를 덤프한 것이다.
(단 이더넷의 Preamble, SFD, CRC 필드는 없고, 목적지 MAC 주소부터 덤프한것임)

```
00 30 6d 99 1c 03 00 90  08 a6 12 5e 08 00 45 00
00 3b 77 0a 00 00 80 11  13 f7 d3 f8 07 bf d3 f8
00 01 04 e1 00 35 00 27  92 27
```

1. 이더넷 PDU

- 1.1 이더넷 PDU헤더는?
- 1.2 목적지 MAC 주소는?
- 1.3. 발신지 MAC 주소는?
- 1.4 이더넷 형태는?
- 1.5 네트워크 계층은 어떤 서비스인가?

2. IP PDU

- 2.1 IP PDU헤더는?
- 2.2 목적지 IP 주소는?
- 2.3. 발신지 IP 주소는?
- 2.4 프로토콜 형태는?
- 2.5 IP의 상위 계층인 전송 계층은 어떤 서비스인가?



2.6 IP 버전은?

2.7 IP 헤더 길이는?

2.3. TTL 값은?

2.4 ID 값은?

2.5 헤더 체크섬값은?

2.6 전체 길이값은?

3. UDP PDU

3.1 UDP PDU헤더는?

3.2 목적지 포트는?

3.3 발신지 포트는?

3.4 UDP 길이는?

3.5 UDP 헤더 체크섬은?

3.6 덤프한 패킷 뒤에는 몇바이트가 추가됨을 캡처하지않았는가?

4. 발신지와 착신지 IP주소를 반대로 하여(즉, 이 패킷 수신후 응답하는)

패킷의 hex값을 이더넷부터 작성하라. (불확실한값은 임의로 넣을것)

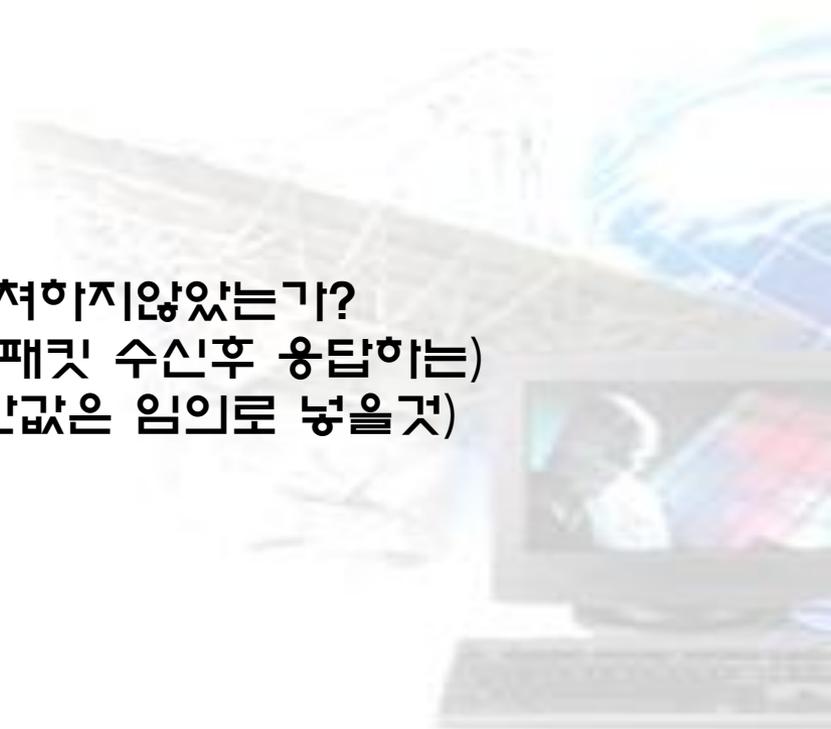
파라미터 값을 설정한 이유도 나열

- IP 헤더 체크섬: Ox2be8

- UDP 헤더 체크섬: Ox82Of

- IP 패킷 길이: 159

- IP 옵션은 없음



정답: 1, 2,3 문제

```
[-] Ethernet II, Src: Hana_a6:12:5e (00:90:08:a6:12:5e), Dst: LucentTe_99:1c:03 (00:30:6d:99:1c:03)
  [+ Destination: LucentTe_99:1c:03 (00:30:6d:99:1c:03)
  [+ Source: Hana_a6:12:5e (00:90:08:a6:12:5e)
    Type: IP (0x0800)
[-] Internet Protocol, Src: 211.248.7.191 (211.248.7.191), Dst: 211.248.0.1 (211.248.0.1)
  Version: 4
  Header length: 20 bytes
  [+ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
  Total Length: 59
  Identification: 0x770a (30474)
  [+ Flags: 0x00
  Fragment offset: 0
  Time to live: 128
  Protocol: UDP (0x11)
  [+ Header checksum: 0x13f7 [correct]
  Source: 211.248.7.191 (211.248.7.191)
  Destination: 211.248.0.1 (211.248.0.1)
[-] User Datagram Protocol, Src Port: mesavistaco (1249), Dst Port: domain (53)
  Source port: mesavistaco (1249)
  Destination port: domain (53)
  Length: 39
  [+ Checksum: 0x9227 [correct]
```

1. 이더넷 PDU

1.1 이더넷 PDU헤더는? 00:30:6d:99:1c:03:00:90:08:a6:12:5e:08 00

1.2 목적지 MAC 주소는? 00:30:6d:99:1c:03

1.3. 발신지 MAC 주소는? 00:90:08:a6:12:5e

1.4 이더넷 형태는? 0800

1.5 네트워크 계층은 어떤 서비스인가? IP 프로토콜로 서로다른 네트워크의 IP 패킷 전달 서비스

2.1 IP PDU헤더는?

```
45 00  
00 3b 77 0a 00 00 80 11 13 f7 d3 f8 07 bf d3 f8  
00 01
```

45 00 00 3c 40 32 00 00 80 01 00 00 dc 45 da f2 dc 44 86 01

2.2 목적지 IP 주소는? 211.248.0.1

2.3. 발신지 IP 주소는? 211.248.7.191

2.4 프로토콜 형태는? UDP(0x11)

2.5 IP의 상위 계층인 전송 계층은 어떤 서비스인가? UDP

2.6 IP 버전는? IPv4

2.7 IP 헤더 길이는? 20 Bytes

2.3. TTL 값은? 128

2.4 ID 값은? 770a

2.5 헤더 체크섬값은? 13f7

2.6 전체 길이값은? 59 (IP헤더 제외시 39바이트)

3. UDP PDU

3.1 UDP PDU헤더는?

```
04 e1 00 35 00 27 92 27
```

3.2 목적지 포트는? 53(DNS)

3.3 발신지 포트는? 1249(임의의 unwell-known port)

3.4 UDP 길이는? 39

3.5 UDP 헤더 체크섬은? 0x9227

3.6 덤프한 패킷 뒤에는 몇바이트가 추가됨을 캡쳐하지않았는가? 31바이트



정답: 4 문제

```
00 90 08 a6 12 5e 00 30 6d 99 1c 03 08 00 45 00
00 9f a0 b4 40 00 fe 11 2b e8 d3 f8 00 01 d3 f8
07 bf 00 35 04 e1 00 8b 82 0f
```

- [-] Ethernet II, Src: LucentTe_99:1c:03 (00:30:6d:99:1c:03), Dst: Hana_a6:12:5e (00:90:08:a6:12:5e)
 - [-] Destination: Hana_a6:12:5e (00:90:08:a6:12:5e)
 - Address: Hana_a6:12:5e (00:90:08:a6:12:5e)
 - 0 = IG bit: Individual address (unicast)
 - 0. = LG bit: Globally unique address (factory default)
 - [-] Source: LucentTe_99:1c:03 (00:30:6d:99:1c:03)
 - Address: LucentTe_99:1c:03 (00:30:6d:99:1c:03)
 - 0 = IG bit: Individual address (unicast)
 - 0. = LG bit: Globally unique address (factory default)
 - Type: IP (0x0800)
- [-] Internet Protocol, Src: 211.248.0.1 (211.248.0.1), Dst: 211.248.7.191 (211.248.7.191)
 - Version: 4
 - Header length: 20 bytes
 - [-] Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
 - Total Length: 159
 - Identification: 0xa0b4 (41140)
 - [-] Flags: 0x04 (Don't Fragment)
 - Fragment offset: 0
 - Time to live: 254
 - Protocol: UDP (0x11)
 - [-] Header checksum: 0x2be8 [correct]
 - [Good: True]
 - [Bad : False]
 - Source: 211.248.0.1 (211.248.0.1)
 - Destination: 211.248.7.191 (211.248.7.191)
- [-] User Datagram Protocol, Src Port: domain (53), Dst Port: mesavistaco (1249)
 - Source port: domain (53)
 - Destination port: mesavistaco (1249)
 - Length: 139
 - [-] Checksum: 0x820f [correct]