Chapter 11

World Wide Web and HTTP

OBJECTIVES:

- WWW 구조
- Web clients and Web servers
- URL 정의
- HTTP 프로토콜

Chapter Outline

11-1 ARCHITECTURE

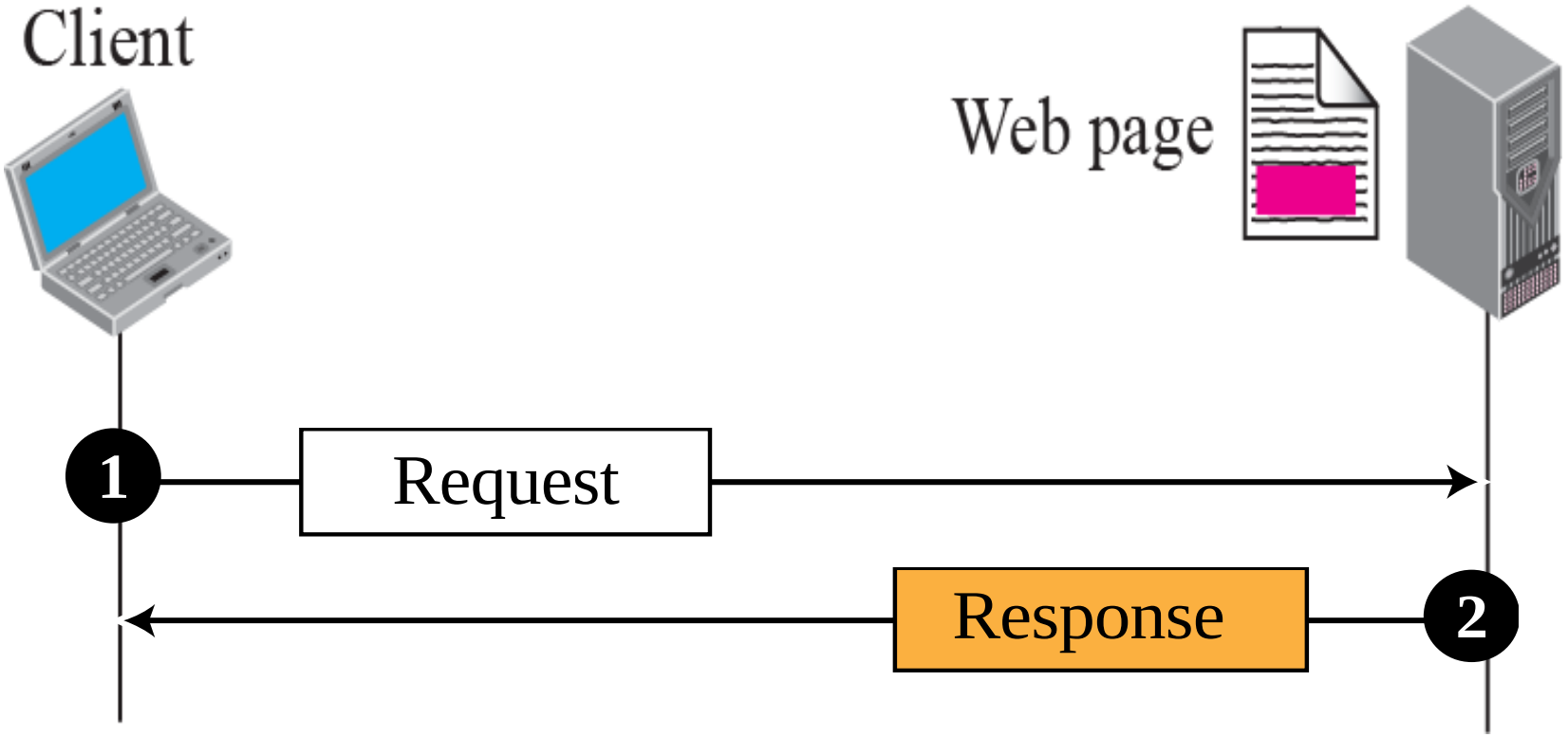
-WWW: distributed client-server service

. Client: browser를 사용하여 서버의 서비스에 접근

. 제공된 서비스는 여러 곳에 위치한 사이트(sites)에 분산되어 Web pages의 한 개 이상 도큐먼트로 지정

. Each Web page는 동일하거나 다른 사이트에 많은 링크로 접속

간단한 문서의 요청과 응답



URL

11-2 HTTP

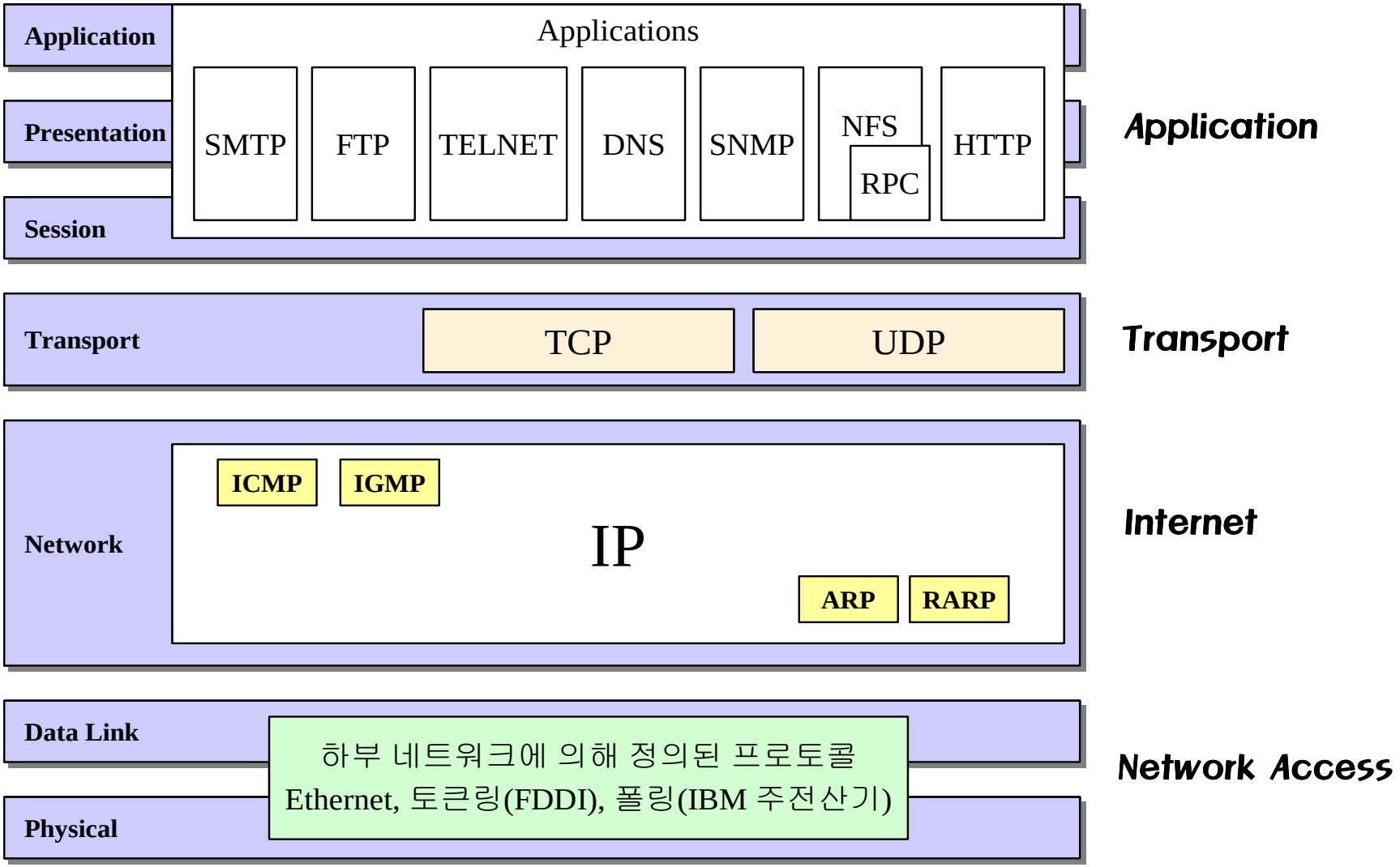
-Hypertext Transfer Protocol (HTTP):

World Wide Web에 접속 하기 위한 프로토콜 .

- HTTP : FTP(포트 20, 21)와 SMTP의 조합

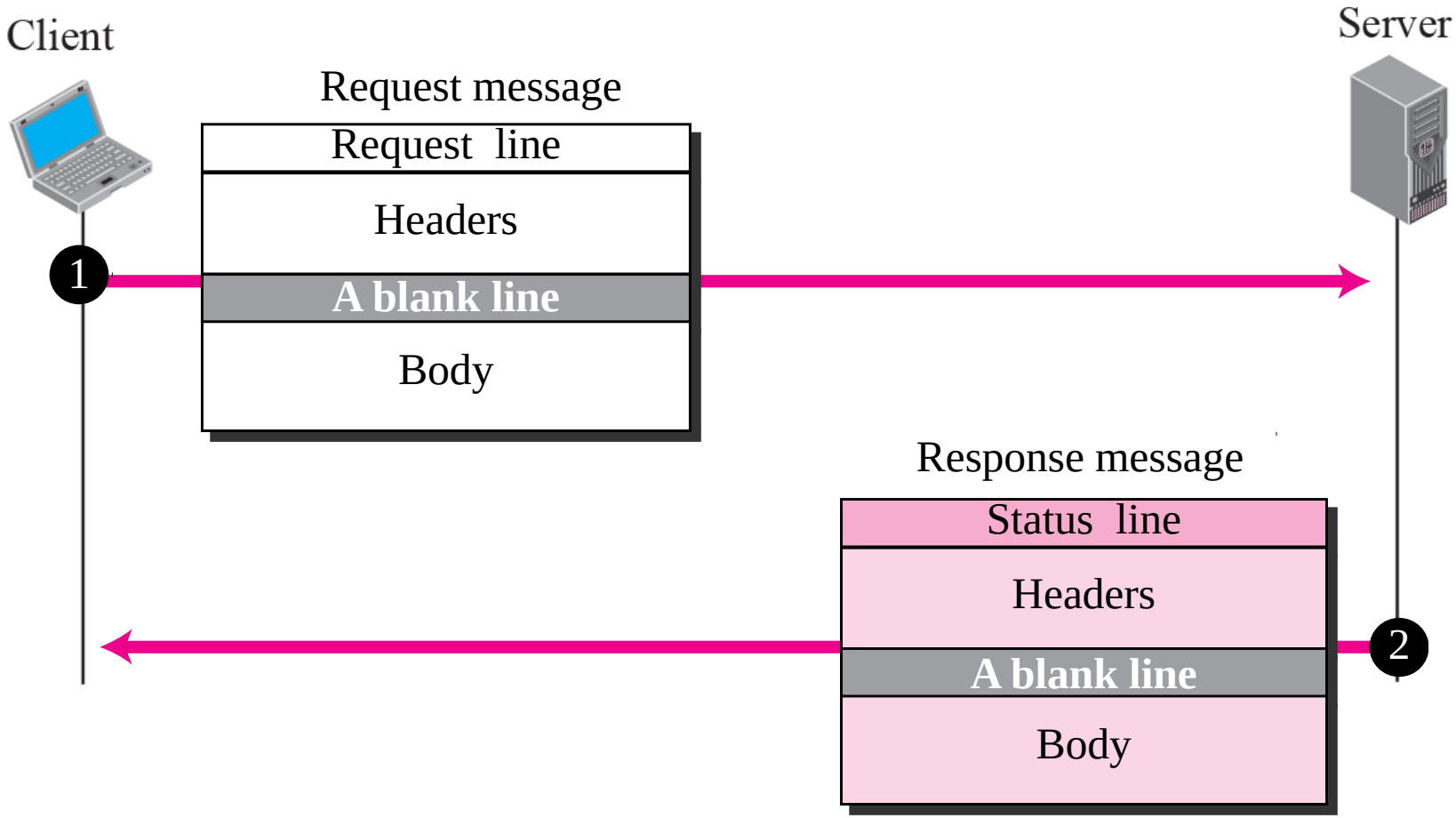
. FTP 관점: 파일을 보내고 TCP 사용, TCP 포트 80 사용하고 별도의 제어 없이 데이터만 클라이언트와 서버 사이에 전달 따라서 FTP보다 좀 단순

. 클라이언트(WWW 브라우저)와 서버(WWW 서버) 사이의 데이터의 양방향 처리(읽혀지고 쓰여짐), SMTP 메시지는 사람이 읽는 단방향성

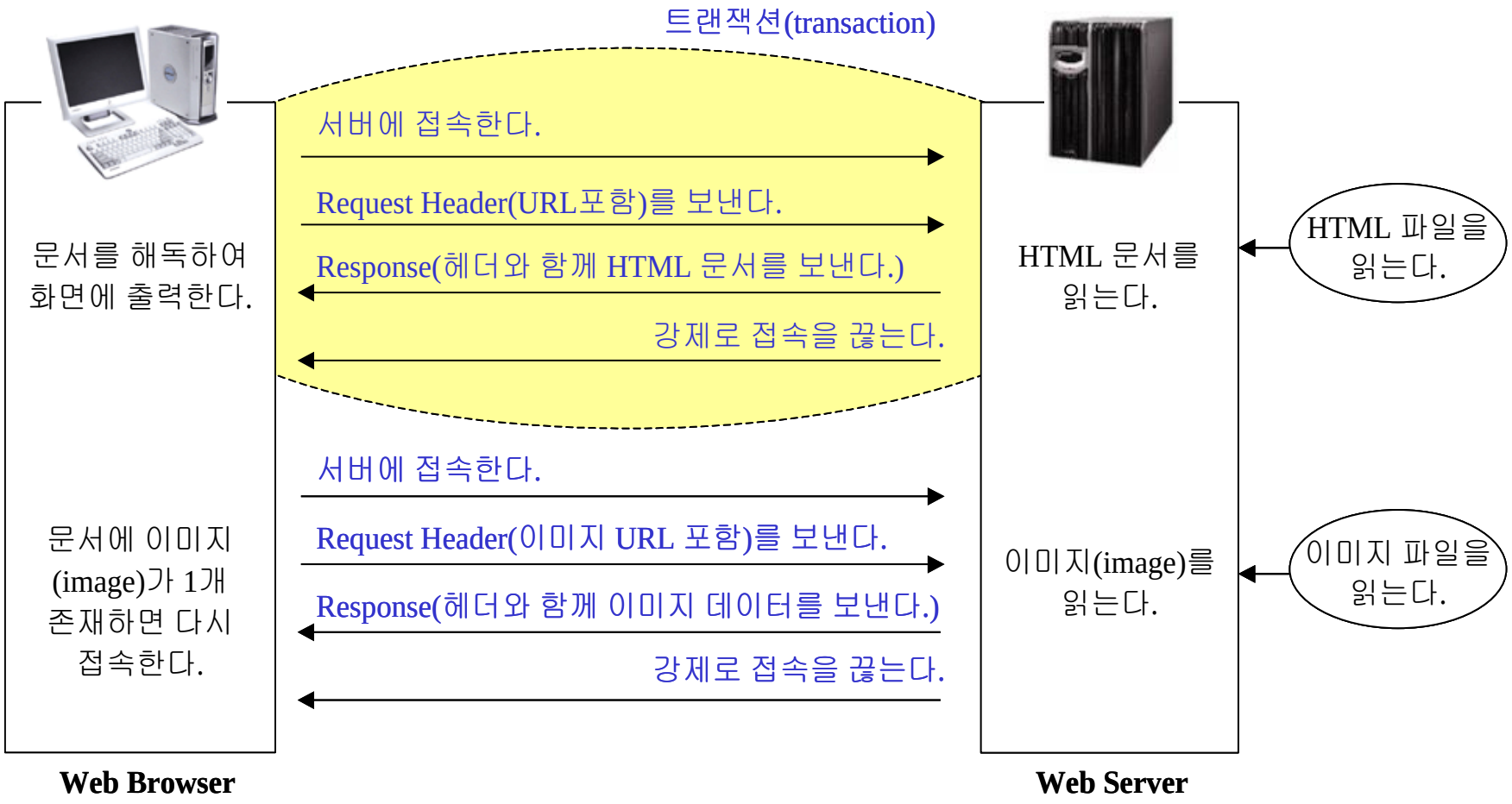

Application
Presentation
Session
Applications
SMTP
FTP
TELNET
DNS
SNMP
NFS
RPC
HTTP
Application
Transport
TCP
UDP
Transport
Network
ICMP
IGMP
IP
ARP
RARP
Internet
Data Link
Physical
하부 네트워크에 의해 정의된 프로토콜
Ethernet, 토큰링(FDDI), 폴링(IBM 주전산기)
Network Access

HTTP transaction:

HTTP transaction: 웹 브라우저가 웹 서버에 하나의 요청을 보내고 웹 서버가 요청을 처리하여 응답을 전송하는 한 번의 과정을 트랜잭션(transaction)



HTTP transaction 처리과정



▪트랜잭션(transaction)

① 웹 브라우저가 웹 서버가 설치된 호스트에 연결한다.

② 웹 브라우저는 요청 패킷(request packet)을 만든다.

③ 웹 브라우저는 요청 패킷(헤더 부분과 데이터 부분)을 전송하고 기다린다.

④ 웹 서버는 요청 패킷을 받고 헤더와 데이터로 분리한다. 웹 서버는 요청을 처리하고 응답 패킷을 만든다.

⑤ 웹 서버는 응답 패킷을 웹 브라우저에게 보내고 강제로 연결을 끊는다.

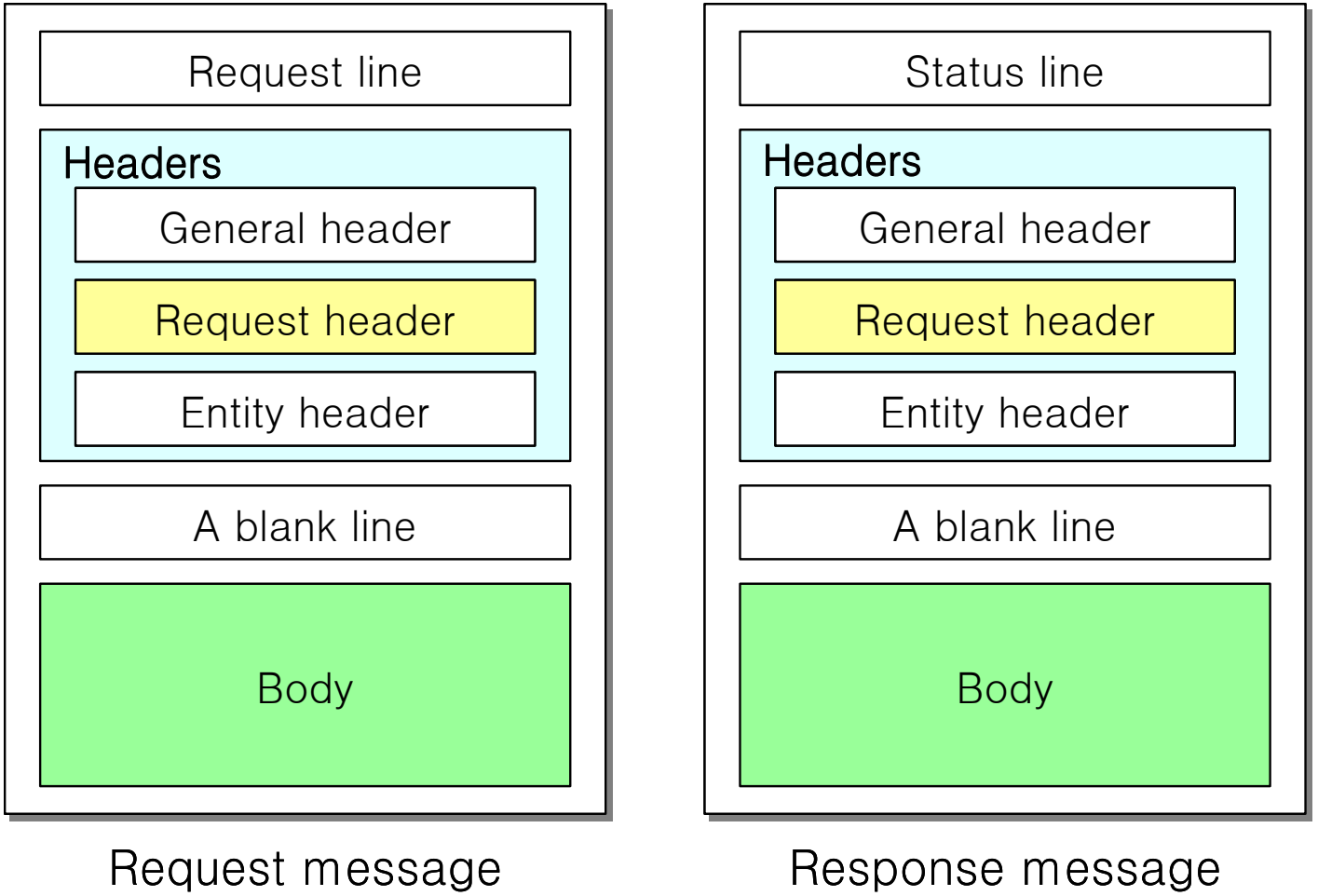
· HTTP 트랜잭션의 특징

연결 당 하나의 트랜잭션을 수행하고 각 트랜잭션은 상호 무관하다.
그러므로, 웹 서버는 연속되는 각 트랜잭션을 독립적인 사건으로 다룬다.

· HTTP 트랜잭션의 장점

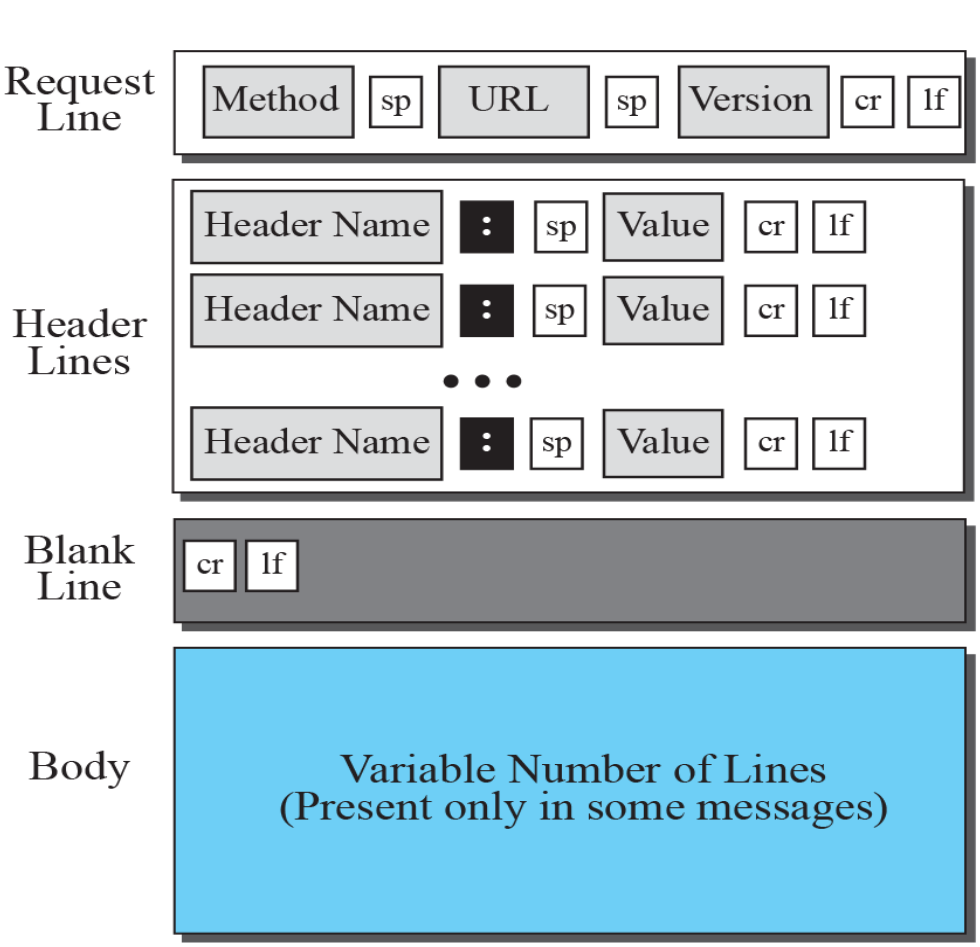
웹 서버의 부하를 줄일 수 있으므로 동시에 많은 클라이언트를 지원할 수 있는 것이다.

Format of the request and response message



요청 메시지(Request) 형태

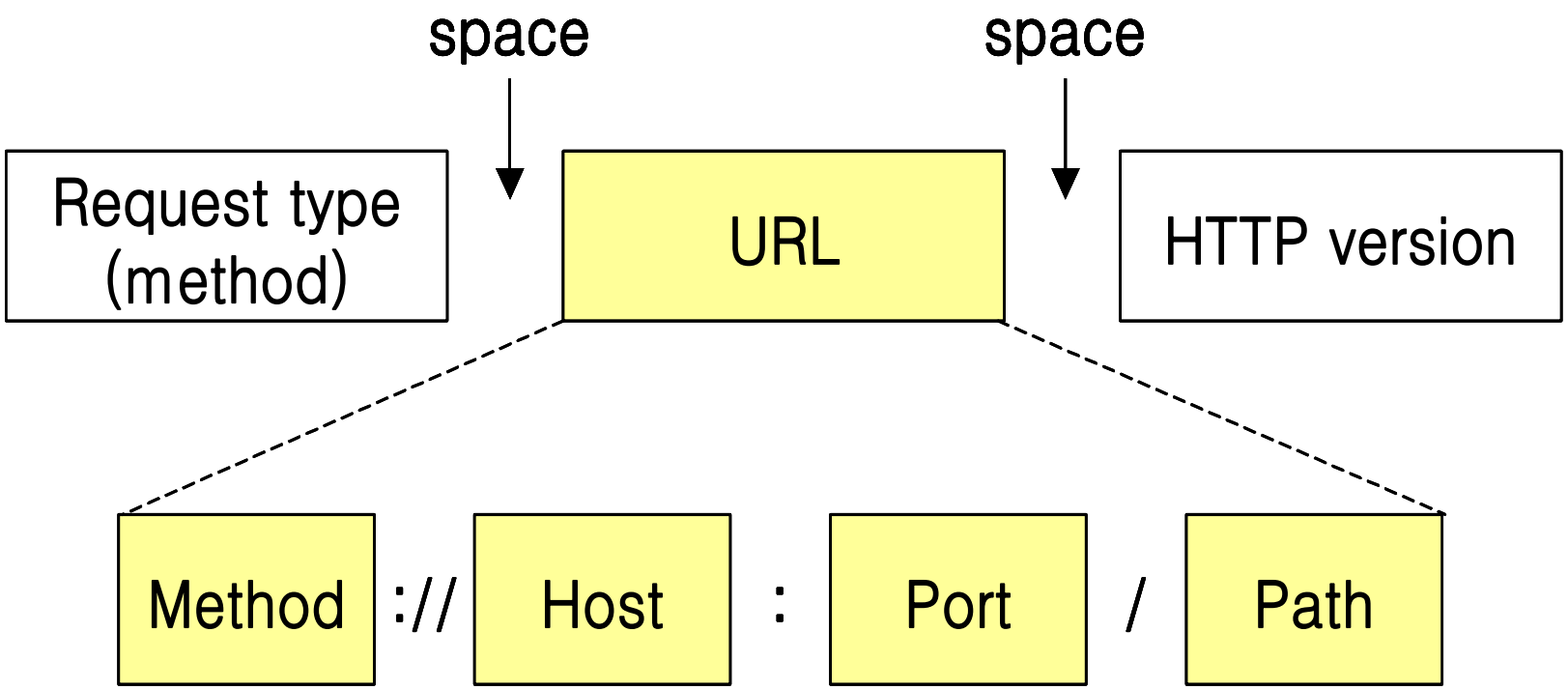
요청 메시지(Request): 요청메시지는 요청 라인(request line), 헤더(header) 그리고 몇몇 경우 본문(body)으로 구성된다.



Legend

sp: Space
cr: Carriage Return
lf: Line Feed

요청 라인**(Request line):** 요청 종류**(request type),** 자원**(URL),** **HTTP** 버전**(version)** 등을 정의한다**.**



▪요청 종류**(Request type)**

여러 가지 요청 유형들이 정의되고 이 유형은 요청 메시지를 여러 가지 메소드로 분류한다**.** 메소드**(Method)**는 클라이언트가 서버에게 보내는 실제 명령이거나 요청이다**.**

▪메소드(Method)

- GET : 클라이언트가 서버로부터 문서를 읽어 오기 원할 때 사용.
  서버는 오류가 없는 경우 보통 응답메시지의 본문에 문서의 내용을 담아서 응답한다.
- HEAD : 문서 자체가 아니라 문서에 대한 어떤 정보를 원할 때 사용,
  이는 GET과 비슷하지만 서버로부터의 응답에 본문이 없는 점이 다르다.
- POST : 클라이언트가 서버에게 어떤 정보를 제공할 때 사용된다. 예를 들어, 서버에게 입력을 보낼 때 사용한다.
- PUT : 클라이언트가 서버에 저장될 새 문서 혹은 교체 문서를 제공하기 위해 사용되고 문서는 요청의 본문에 포함
  되고 URL에 의해 지정된 위치에 저장될 것이다.
- PATCH : PUT과 비슷하나 요청이 기존 파일에서 구현되어야 하는 변경사항의 목록만을 포함하고 있다는 점이 다르다.
- COPY : 파일을 다른 위치로 복사하기 위해 사용되고 원본파일의 위치는 요청 라인(URL)에서 주어지고 목적지의
  위치는 항목 헤더에서 주어진다.
- MOVE : 파일을 다른 위치로 이동하기 위해 사용되고, 원본파일의 위치는 요청 라인(URL)에서 주어지고 목적지의
  위치는 항목 헤더에서 주어진다.
- DELETE : 서버에서 문서를 제거하기 위해 사용된다.
- LINK : 문서에서 다른 위치로 링크나 링크들을 생성하기 위해 사용되고, 원본파일의 위치는 요청 라인(URL)에서
  주어지고 목적지의 위치는 항목 헤더에서 주어진다.
- UNLINK : LINK 메소드에 의해 생성된 링크를 삭제하기 위해 사용된다.
- OPTION : 클라이언트가 서버에게 사용 가능한 옵션을 질의하기 위해 사용한다.

URL

웹 페이지를 액세스하기 원하는 클라이언트는 주소를 필요로 한다. 전 세계에 퍼져있는 문서들에 대한 액세스를 가능하게 하기 위하여, **HTTP**는 위치 지정자**(locator)**라는 개념을 사용한다.

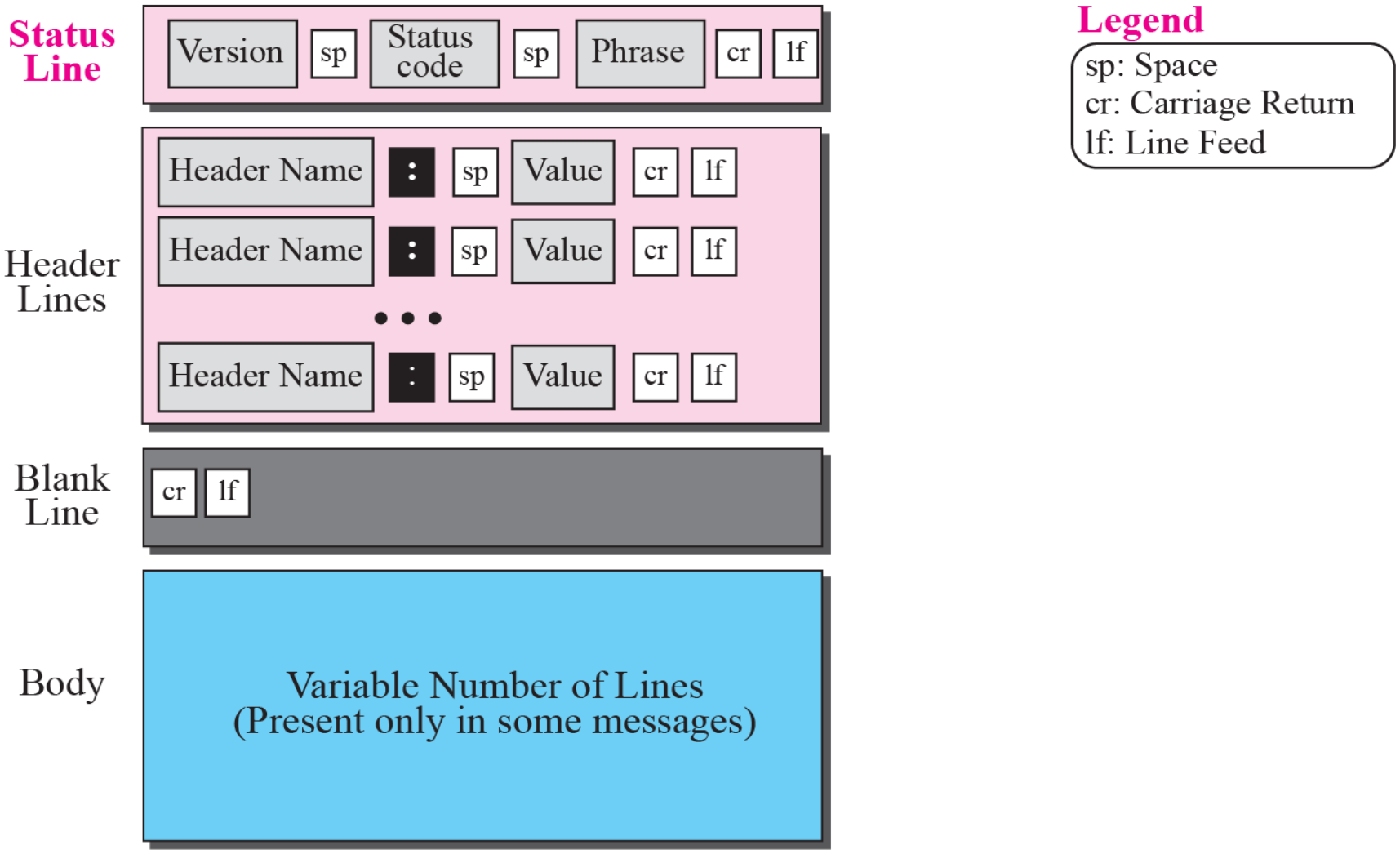
- 프로토콜 **:** 문서를 가져오기 위해 사용되는 프로토콜을 명시하는 것으로 **FTP, HTTP, TELNET** 등의 프로토콜 중 하나를 지정한다.
- 호스트 주소 **:** 요청한 정보가 위치해 있는 시스템의 주소**(IP or Domain name)**를 나타낸다.
- 포트번호 **: URL**은 접근하는 서버의 **port** 번호를 지정할 수 있다.

·버전**(Version)**

HTTP의 버전을 명시한다. 대부분 **1.1**을 사용한다.

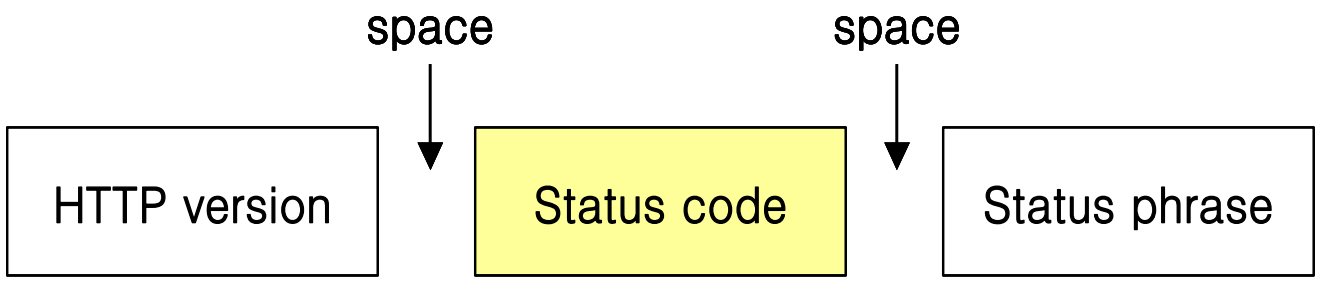
응답 메시지(Response) 형태

응답 메시지는 상태 라인, 헤더, 그리고 몇몇 경우 본문(Body)으로 구성된다.



상태 라인(Status line):

상태 라인은 HTTP 버전, 상태 코드, 상태 설명



- **HTTP version**

요청 라인의 필드와 동일하다.

- **상태 코드(Status code)**

FTP와 SMTP 프로토콜에 있는 것과 유사하다. 이는 세 자리 숫자로 구성된다.
하위 두 개의 십진수는 범주를 구분하는데 사용되고, 첫 번째 십진수는 5가지 범주로 나뉜다.
HTTP 상태 코드는 확장 가능한 승인된 코드이다.

- **상태 설명(Status phrase)**

문자 형태로 되어 있으며 상태 코드를 설명한다.

상태코드의 첫 번째 십진수 응답 분류

십진수	의 미
1xx	정보 - HTTP/1.0에서는 사용되지 않고 HTTP/1.1에서 상태 라인과 추가적인 헤더를 통해 클라이언트로 임시적인 응답을 전달
2xx	성공 - 요청이 성공적으로 수락되었고 이해되었음
3xx	재전송 요구 - 요청을 완성하기 위해서는 더 많은 정보 필요
4xx	클라이언트 에러 - 요청에 잘못된 문법을 사용하였거나 실행할 수 없음
5xx	서버 에러 - 서버가 타당한 요청을 받았지만 수행시 실패

상태코드

코드	의미	설 명
200	OK	요청이 성공적으로 수행되었음
201	Created	요청이 새롭게 생성된 자원에 의해 이행됨
202	Accepted	요청은 받아들여졌지만 완전하게 수행되지 않았음
204	No Content	서버는 요청을 이행했지만 전달할 정보가 없는 경우
301	Moved Permanently	요청한 자원은 새로운 URL이 할당되었고, 해당 리소스를 사용하기 위해서는 변경된 URL을 사용하여 접속해야 하는 경우
302	Found	요청한 자원이 일시적으로 다른 URL로 옮겨진 경우
304	Not Modified	클라이언트가 조건적인 GET 명령을 수행하기 위해 URL로 접근하였으나 If-Modified-Since 필드의 날짜가 경과하여 데이터를 사용할 수 없음
400	Bad Request	클라이언트로부터 입력된 요청이 서버측에서 이해할 수 없는 요청인 경우
401	Unauthorized	클라이언트 인증을 위한 요청을 전달하였으나 적합하지 않은 인증 데이터를 사용한 경우
403	Forbidden	서버는 클라이언트의 요청을 이해하였으나 수행이 거부된 경우
404	Not Found	서버가 요청한 URL을 찾지 못한 경우
500	Internal Server Error	서버가 요청을 수행할 수 없는 상태에 있는 경우
501	Not Implemented	서버가 요청을 수행하지 않는 경우
502	Bad Gateway	게이트웨이 역할을 하는 서버가 상위 서버로부터 사용할 수 없는 응답을 받은 경우
503	Service Unavailable	일시적인 서버의 부하나 유지/관리상의 문제로 요청을 처리할 수 없는 상태인 경우

헤더 라인은 일반**(General),** 요청**(Request),**응답**(Response),**
항목**(Entity)** 헤더의 **4**가지 중 하나에 속하게 된다**.**
요청 메시지는 일반**,** 요청**,** 항목 헤더만 포함할 수 있고**,**
응답 메시지는 일반**,** 응답**,** 항목 헤더만 포함할 수 있다**.**

일반 헤더(General):메시지에 대한 일반적인 정보를 제공하며 요청과 응답에 모두 존재할 수 있다

일반 헤더 종류 및 의미

헤더명	설 명
Cache-Control	캐시 메커니즘에 대한 설정
Connection	연결 설정에 대해 요구사항을 전달
Date	날짜 전달
Pragma	요청과 응답 과정에서 적용되는 특정 명령 전달
Transfer-Encoding	서버와 클라이언트 사이에 메시지를 안전하게 전달하기 위해 변형된 인코딩 타입을 지정
Upgrade	통신시에 추가로 지원되어야 할 프로토콜 정보 전달
Via	게이트웨이나 프락시에서 경로상에 사용되는 프로토콜 명시

요청 헤더(Request): 요청 메시지에만 존재하고 이는 클라이언트 구성과 클라이언트가 선호하는 문서형식을 지정한다.

요청 헤더 종류 및 의미

헤더명	설 명
Accept	응답 메시지 수락시 미디어 타입을 명시
Accept-Charset	응답시 수락되는 문자 셋(set)을 나타냄
Accept-Encoding	Accept와 유사 응답 메시지 수락시 압축을 통한 코딩율을 제한
Accept-Language	Accept와 유사 사용되는 언어의 제한
Authorization	사용자의 인증 절차 수행시
From	요청 에이전트를 제어하는 사용자의 인터넷 메일 주소
Host	요청된 자원의 호스트 명과 포트번호
If-Modified-Since	GET 메소드와 함께 사용되는 조건문으로 명시한 날짜 이후의 자료이면 전달
If-Match	메소드와 함께 사용되는 조건문으로 이전에 획득한 엔티티 중 조건이 맞는 것이 있으면 사용하고 그렇지 않으면 서버를 통해 다시 획득하는 cache 기능 수행
If-None-Match	메소드와 함께 사용되는 조건문으로 이전에 획득한 엔티티 중 조건이 맞는 것이 없으면 서버를 통해 획득하는 cache 기능 수행
If-Unmodified-Since	메소드와 함께 사용되는 조건문으로 요청된 자원이 명시한 날짜 이후에 작성하지 않았다면 기능을 수행
Max-Forwards	TRACE 메소드와 함께 사용되는 것으로 게이트웨이의 수를 제한할 때 사용
Proxy-Authorization	클라이언트가 proxy로 요청메시지 전달시 클라이언트의 인증이 필요할 때 사용
Range	HTTP 메시지의 표현 형식 지정
Referer	다양한 정보의 전달(자원 전달 경로, 최적화된 cache 정보, log 정보 등)
User-Agent	요청하는 사용자 에이전트에 대한 정보 전달

응답 헤더(Response Header)

응답 메시지에만 존재할 수 있고 서버의 구성과 요청에 대한 특별한 정보를 지정한다.

응답 헤더 종류 및 의미

헤더명	설 명
Age	서버가 응답을 시작한 시간부터 측정한 시간 정보
Location	자원 확보한 위치 정보 전달
Proxy-Authenticate	Proxy 인증시 사용
Public	서버에서 지원하는 메소드의 리스트 전달
Retry-After	클라이언트가 해당 자원을 언제까지 사용할 수 없는지에 대한 정보 전달
Server	서버가 사용하는 소프트웨어 정보
Warning	응답 상태에 대한 추가적인 정보 전달. 주로 캐싱 기능 수행시 의미 전달 부족시에 전달하는 주의 정보
WWW-Authenticate	응답 코드 401번과 함께 동작. 인증절차에 사용
Accept-Ranges	서버가 요청으로 받아들이는 자원의 정렬 방식

항목 헤더(Entity):

항목 헤더 종류 및 의미

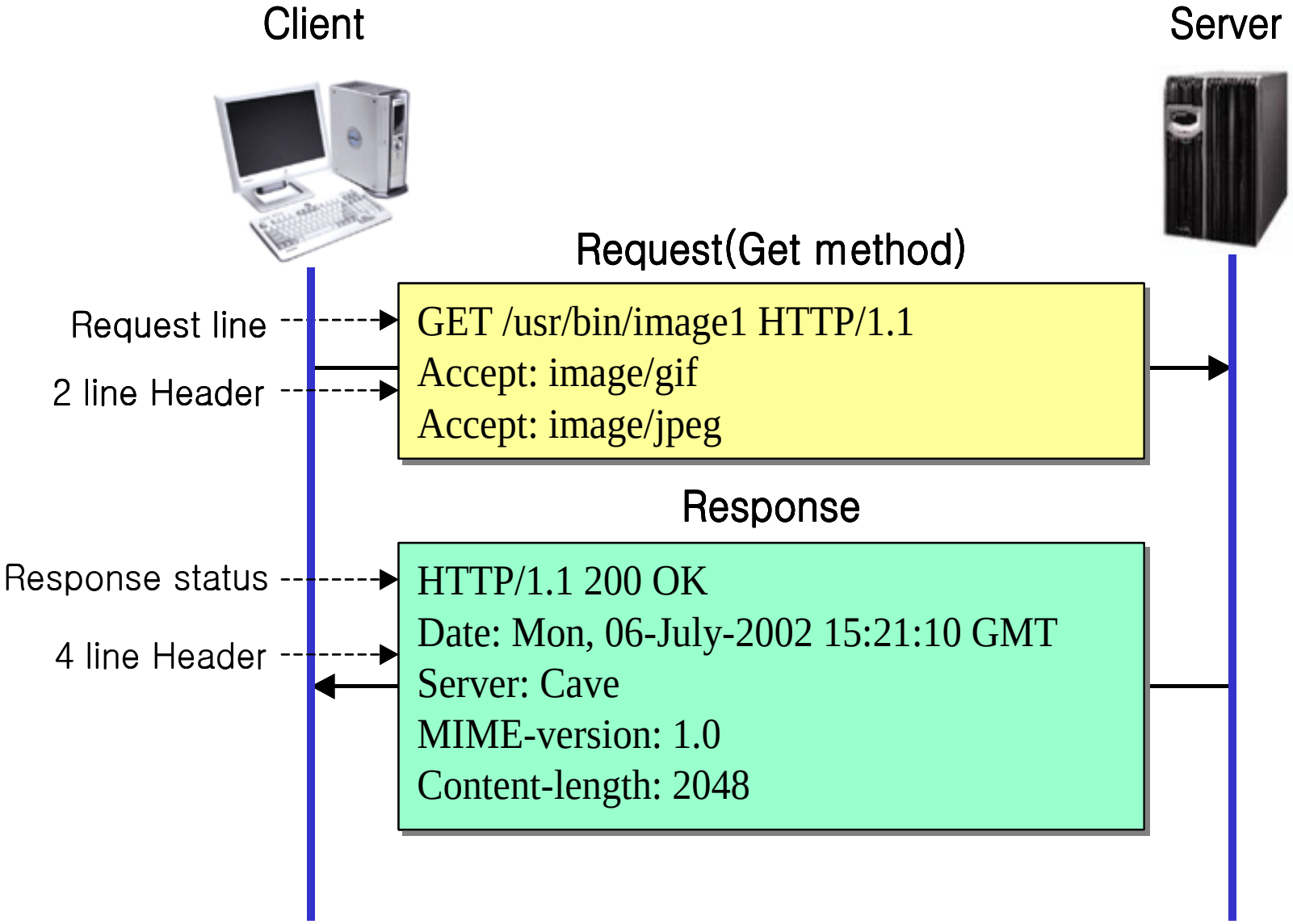
헤더명	설 명
Allow	요청한 URL의 시스템에서 지원할 수 있는 메소드
Content-Base	관련 URL을 해석할 때 사용되는 필드
Content-Encoding	본문(entity)에 사용되는 인코딩 방식을 명시
Content-Language	본문에 사용되는 언어를 나타냄
Content-Length	본문의 길이
Content-Location	본문의 위치 정보
Content-MD5	end-to-end간의 메시지 통합 확인을 위한 요약 정보
Content-Range	전체 본문 중 부분적인 응답을 전달할 경우 부분 메시지 정보 전달
Content-Type	본문에서 사용되는 미디어 타입을 나타냄
ETag	관련 엔티티에서 사용되는 태그
Expires	목적지에서의 응답이 만료된 때의 시간 정보 전달
Last-Modified	자료가 마지막으로 변경된 시점을 전달

Example 11.1

문서를 읽어오는 것으로 경로 **/usr/bin/image1/**에 있는 이미지 파일을 읽어오기 위해 **GET** 메소드를 사용한다**.**

요청 메시지에서는 요청 라인과 **2**개의 헤더를 포함하는데 요청 라인은 **(GET), URL, HTTP** 버전**(1.1)**을 표시하고 헤더는 클라이언트가 **GIF, JPEG**형식으로 화상을 수용할 수 있음을 나타낸다**.** 요청은 본문을 가지고 있지 않다**.**

응답 메시지는 상태 라인과 **4**줄의 헤더를 포함하는데 **4**줄의 헤더 라인에는 날짜**(Date),** 서버**(Server), MIME** 버전**,** 문서길이를 정의한다**.**


Client
Server
Request(Get method)
Request line
2 line Header
GET /usr/bin/image1 HTTP/1.1
Accept: image/gif
Accept: image/jpeg
Response
Response status
4 line Header
HTTP/1.1 200 OK
Date: Mon, 06-July-2002 15:21:10 GMT
Server: Cave
MIME-version: 1.0
Content-length: 2048

Example 11.2

다음 환경의 자기 **PC**에서 구글(**www.google.co.kr**) 접속 시 **ARP, DNS** 질의와 응답 패킷으로 수신된 **IP** 주소로 **TCP**접속, **http** 데이터 전달 과정의 프로토콜과 함께 캡쳐하고 분석하라

① ipconfig /all (나의 IP, MAC, 디폴트 게이트웨이, DNS 서버 주소 확인)

```
Ethernet adapter 로컬 영역 연결:

        Connection-specific DNS Suffix  . :
        Description . . . . . . . . . . . : Intel(R) 82579LM Gigabit Network Con
nection
        Physical Address. . . . . . . . . : 78-2B-CB-AF-6F-AB
        Dhcp Enabled. . . . . . . . . . . : No
        IP Address. . . . . . . . . . . . : 220.69.218.233
        Subnet Mask . . . . . . . . . . . : 255.255.255.128
        Default Gateway . . . . . . . . . : 220.69.218.254
        DNS Servers . . . . . . . . . . . : 220.68.134.1
                                            168.126.63.1
```

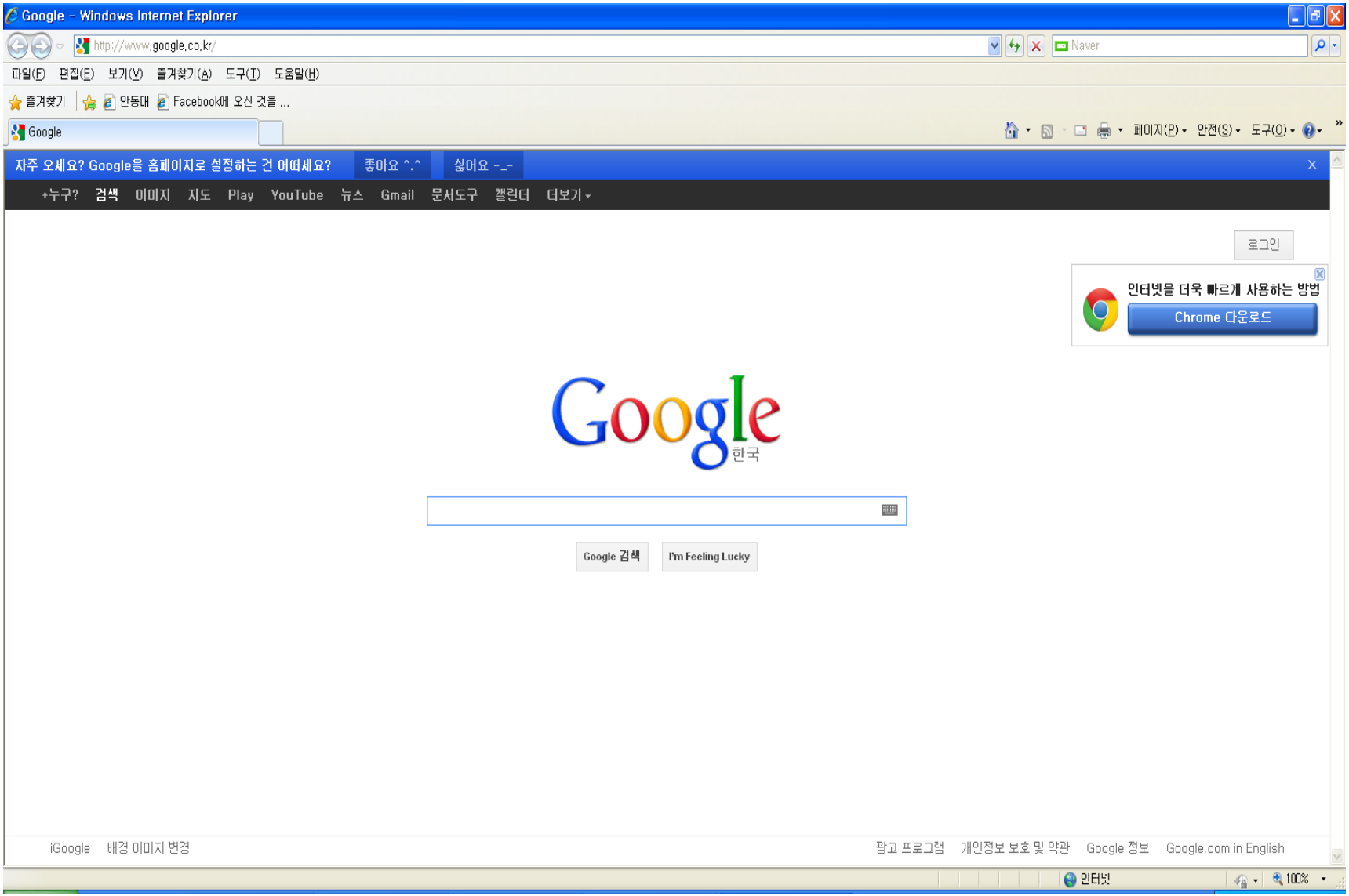
나의 IP 주소
MAC 주소
디폴트게이트 주소
DNS 서버 주소

② ARP 확인 (시작 전 ARP 목록이 있는지 확인을 한 후 목록 삭제)

```
C:₩Documents and Settings₩Administrator>arp -a

Interface: 220.69.218.233 --- 0x2
  Internet Address      Physical Address      Type
  220.69.218.254        30-e4-db-9b-bb-c6     dynamic

C:₩Documents and Settings₩Administrator>arp -d

C:₩Documents and Settings₩Administrator>arp -a
No ARP Entries Found
```

Arp –a : arp 목록 확인
Arp –d : arp 목록 삭제
-> ARP목록 없음

③ 구글 사이트(www.google.com) 접속



④ ARP Request 패킷
(구글의 MAC 주소를 모르기 때문에 ARP 요청으로 MAC 주소 확인)

```
44 51.074119  Dell_af:6f:ab   Broadcast     ARP   42 Who has 220.69.218.254?  Tell 220.69.218.233
45 51.075453  Cisco_9b:bb:c6  Dell_af:6f:ab ARP   60 220.69.218.254 is at 30:e4:db:9b:bb:c6

⊞ Frame 44: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
⊟ Ethernet II, Src: Dell_af:6f:ab (78:2b:cb:af:6f:ab), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  ⊞ Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  ⊞ Source: Dell_af:6f:ab (78:2b:cb:af:6f:ab)
    Type: ARP (0x0806)
⊟ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    [Is gratuitous: False]
    Sender MAC address: Dell_af:6f:ab (78:2b:cb:af:6f:ab)
    Sender IP address: 220.69.218.233 (220.69.218.233)
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 220.69.218.254 (220.69.218.254)

0000  ff ff ff ff ff ff 78 2b  cb af 6f ab 08 06 00 01   ......x+ ..o.....
0010  08 00 06 04 00 01 78 2b  cb af 6f ab dc 45 da e9   ......x+ ..o..E..
0020  00 00 00 00 00 00 dc 45  da fe                     .......E ..
```

ARP Request 헤더

Destination address	Source address	Type	Data	CRC
ff:ff:ff:ff:ff:ff	78:2b:cb:af:6f:ab	0x0806		

0x0001(이더넷타입)		0x0800(이더넷프로토콜)
0x06 (이더넷길이)	0x04 (IP길이)	0x0001 (Request)
78:2b:cb:af:6f:ab(송신지 MAC 주소)		
220.69.218.233(송신지 IP 주소)		
00:00:00:00:00:00(목적지 MAC 주소)		
220.69.218.254(목적지 IP 주소)		

⑤ ARP Reply 패킷 (구글이 외부네트워크이기 때문에 나의 라우터 (디폴트 게이트웨이)의 MAC 주소의 reply)

```
44 51.074119 Dell_af:6f:ab  Broadcast  ARP  42 Who has 220.69.218.254? Tell 220.69.218.233
45 51.075453 Cisco_9b:bb:c6 Dell_af:6f:ab ARP  60 220.69.218.254 is at 30:e4:db:9b:bb:c6
46 51.075468 220.69.218.233 220.68.134.1  DNS  79 Standard query A www.quickzone.co.kr
```

```
⊞ Frame 45: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
⊟ Ethernet II, Src: Cisco_9b:bb:c6 (30:e4:db:9b:bb:c6), Dst: Dell_af:6f:ab (78:2b:cb:af:6f:ab)
  ⊞ Destination: Dell_af:6f:ab (78:2b:cb:af:6f:ab)
  ⊞ Source: Cisco_9b:bb:c6 (30:e4:db:9b:bb:c6)
    Type: ARP (0x0806)
    Trailer: 00000000000000000000000000000000000000
⊟ Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    [Is gratuitous: False]
    Sender MAC address: Cisco_9b:bb:c6 (30:e4:db:9b:bb:c6)
    Sender IP address: 220.69.218.254 (220.69.218.254)
    Target MAC address: Dell_af:6f:ab (78:2b:cb:af:6f:ab)
    Target IP address: 220.69.218.233 (220.69.218.233)
```

```
0000  78 2b cb af 6f ab 30 e4  db 9b bb c6 08 06 00 01   x+..o.0. ........
0010  08 00 06 04 00 02 30 e4  db 9b bb c6 dc 45 da fe   ......0. .....E..
0020  78 2b cb af 6f ab dc 45  da e9 00 00 00 00 00 00   x+..o..E ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

ARP Reply 헤더

Destination address	Source address	Type	Data	CRC
78:2b:cb:af:6f:ab	30:e4:db:9b:bb:c6	0x0806		

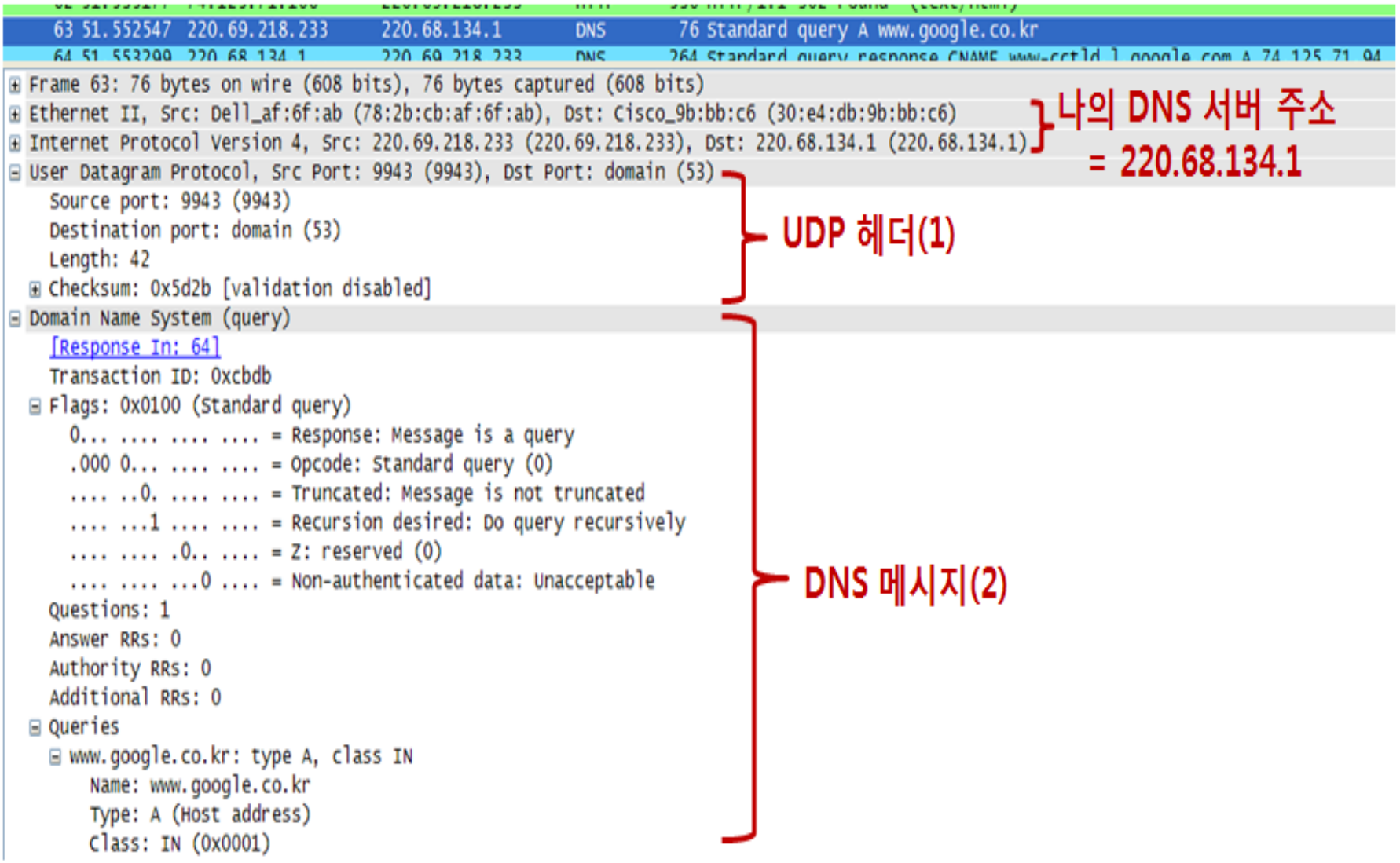
0x0001(이더넷타입)		0x0800(이더넷프로토콜)
0x06 (이더넷길이)	0x04 (IP길이)	0x0002 (Request)
30:e4:db:9b:bb:c6(송신지 MAC 주소)		
220.69.218.254(송신지 IP 주소)		
78:2b:cb:af:6f:ab(목적지 MAC 주소)		
220.69.218.233 (목적지 IP 주소)		

⑥ ARP 목록 확인(ARP 목록에 나의 라우터 주소(디폴트 게이트웨이)가 추가됨)

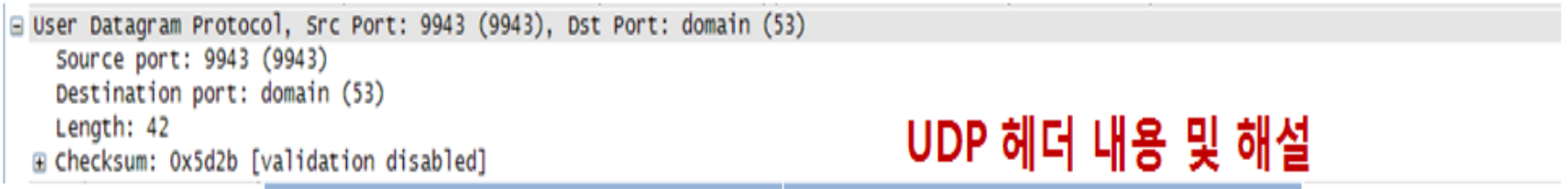
```
C:₩Documents and Settings₩Administrator>arp -a
No ARP Entries Found

C:₩Documents and Settings₩Administrator>arp -a

Interface: 220.69.218.233 --- 0x2
  Internet Address      Physical Address      Type
  220.69.218.254        30-e4-db-9b-bb-c6     dynamic
```

ARP의 요청(Request)와 응답(Reply)로 나의 라우터의 MAC 주소가 ARP 리스트에 추가됨.

⑦ DNS Query 패킷(구글에 접속하기 위해 DNS 서버에 구글의 IP 주소를 물어봄. UDP 패킷에 DNS 메시지 포함됨.)



⑦(1) DNS Query 패킷(UDP 헤더)



UDP 헤더 내용 및 해설

Source port number **0x26d7** (나의 포트번호는 9943으로 임의로 설정된다, 2 bytes)	**Destination port number** **0x0035** (목적지 포트번호는 53(domain)으로 DNS 서버 데이터 요청이다, 2 bytes)
Total length **0x002a** (총 길이는 42이다, 2 bytes)	**Checksum** **0x5d2b** (2 bytes)

⑦(2) DNS Query 패킷(DNS 메시지 요청(query))

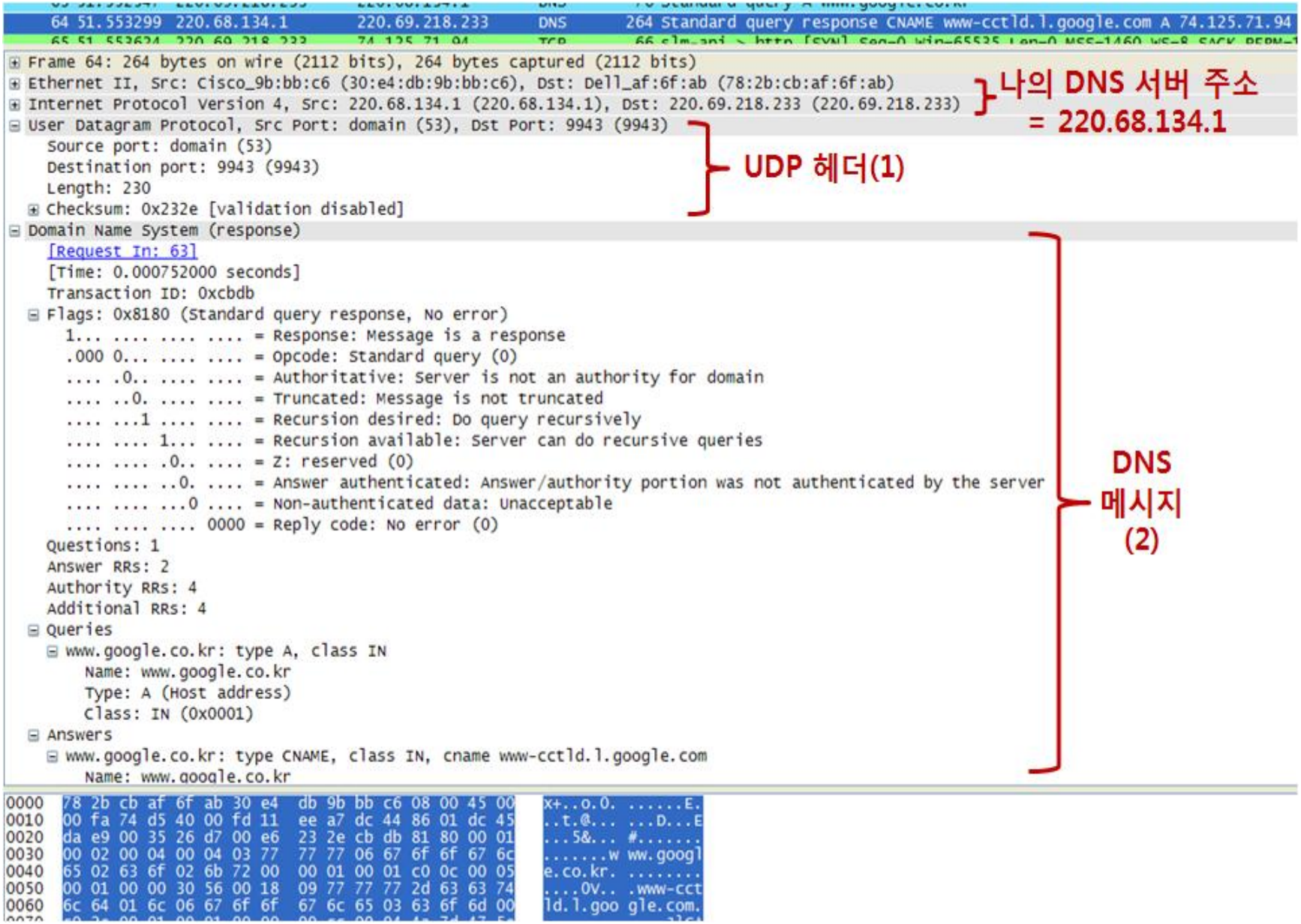
```
Domain Name System (query)
    [Response In: 64]
    Transaction ID: 0xcbdb
  Flags: 0x0100 (Standard query)
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  Queries
    www.google.co.kr: type A, class IN
        Name: www.google.co.kr
        Type: A (Host address)
        Class: IN (0x0001)
```

0xcbdb (나의 ID로 cbdb로 임의 설정)	0x0100 (플래그)
0x0001 (질의 레코드의 수)	0x0000 (응답 레코드의 수)
0x0000 (권한 레코드의 수)	0x0000 (추가 레코드의 수)
Domain name: 3 'w' 6 'g' 'g' 'l' 'c' 'o' 'r' 0	'w' 'w' 'o' 'o' 'e' 2 2 'k'
	0x0001 (쿼리 타입으로 1(A)이다)
0x0001 (쿼리 클래스로 1(IN)이다)	

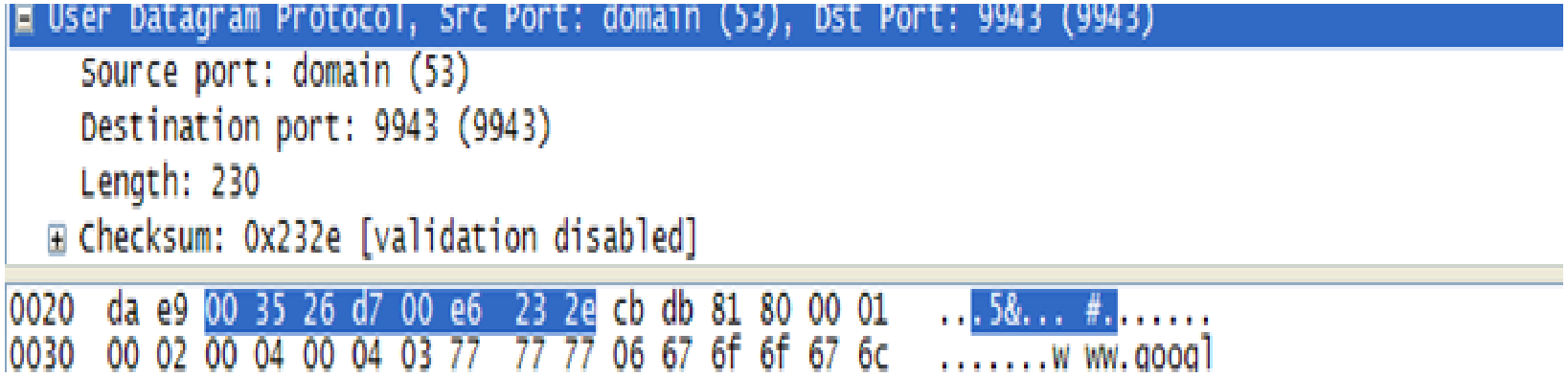
QR	OpCode				AA	TC	RD	RA	Reserved			rCode			
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

플래그 필드

⑧ DNS Response 패킷
(DNS 서버에서 구글의 IP 주소를 찾아 나에게 응답(Response)을 준다)



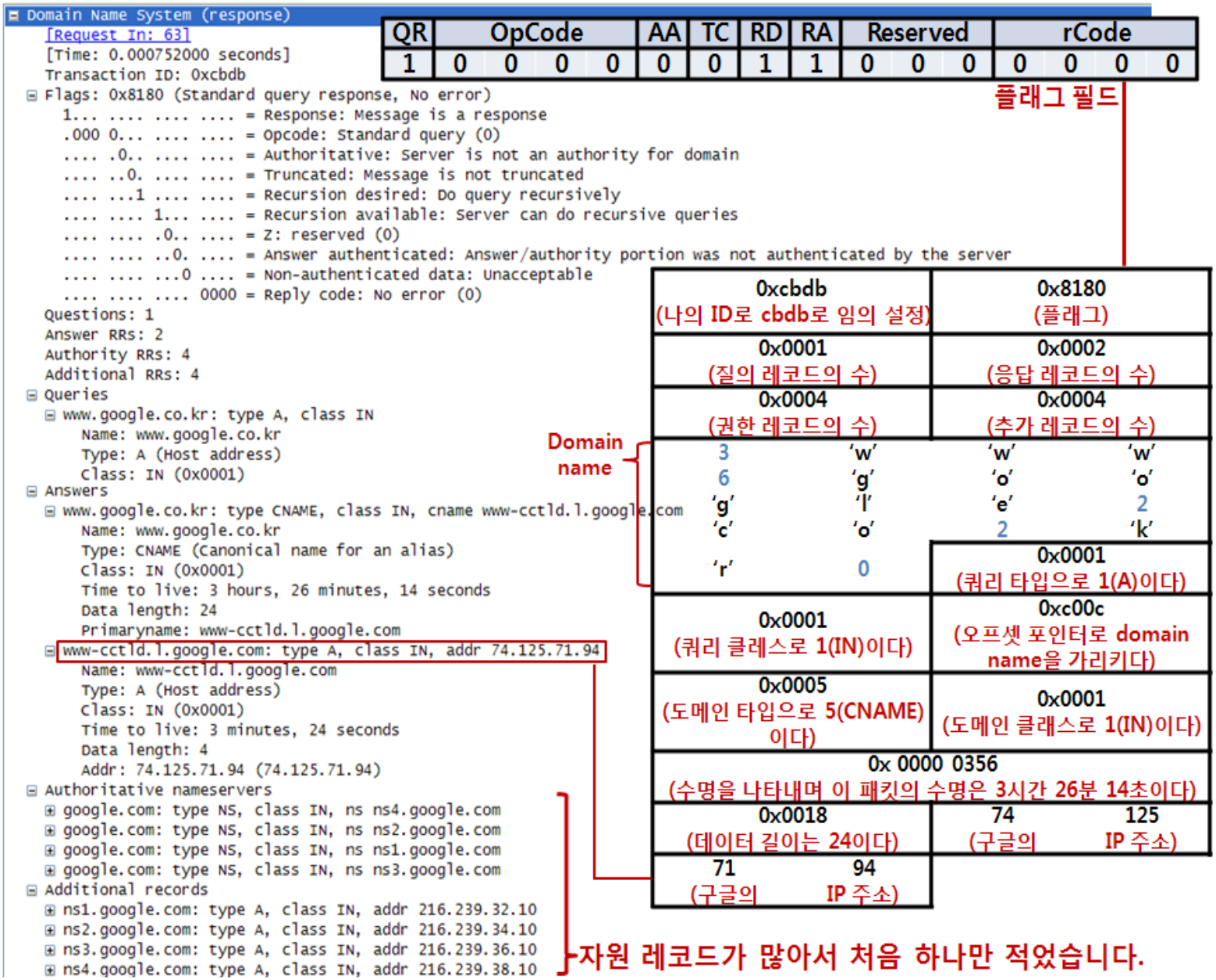
⑧(1) DNS Response 패킷(UDP 헤더)



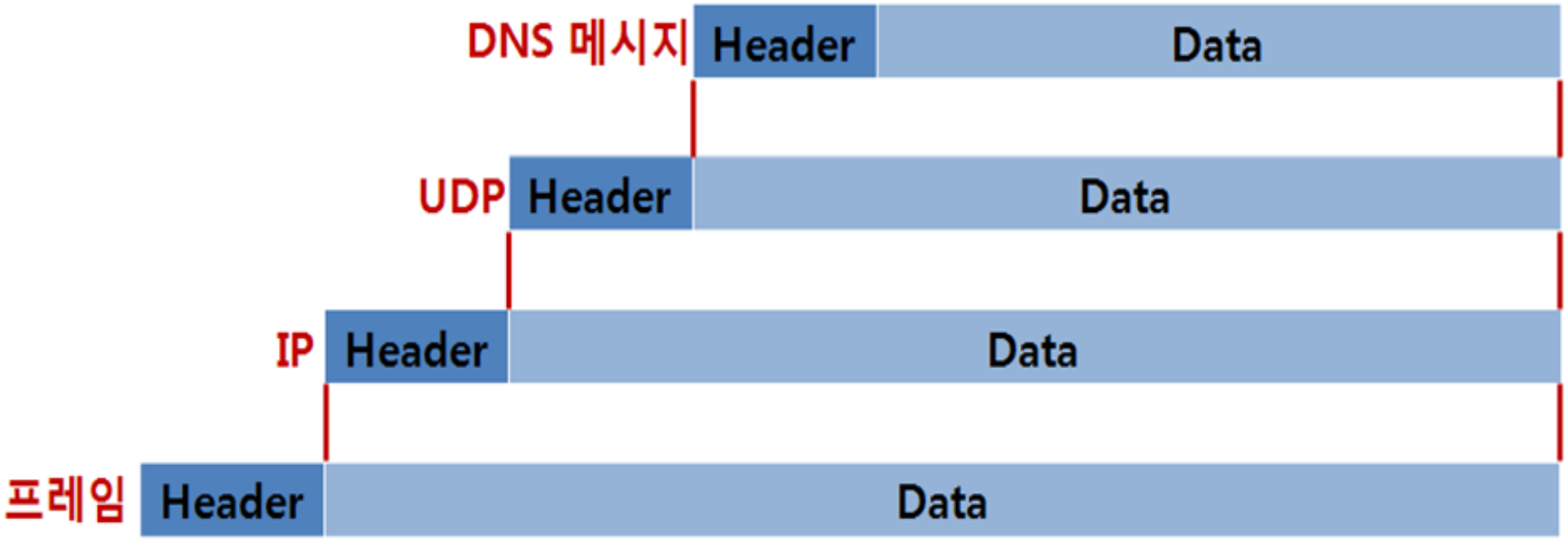
Source port number	Destination port number
0x0035 (송신지 포트번호는 53(domain)으로 DNS 서버의 응답이다, 2 bytes)	0x26d7 (목적지 포트번호는 나의 포트번호인 9943으로 임의 설정이다, 2 bytes)
Total length	**Checksum**
0x00e6 (총 길이는 230이다)	0x23e2 (2 bytes)

UDP 헤더 내용 및 해설

⑧(2) DNS Response 패킷(DNS 메시지 응답(response))

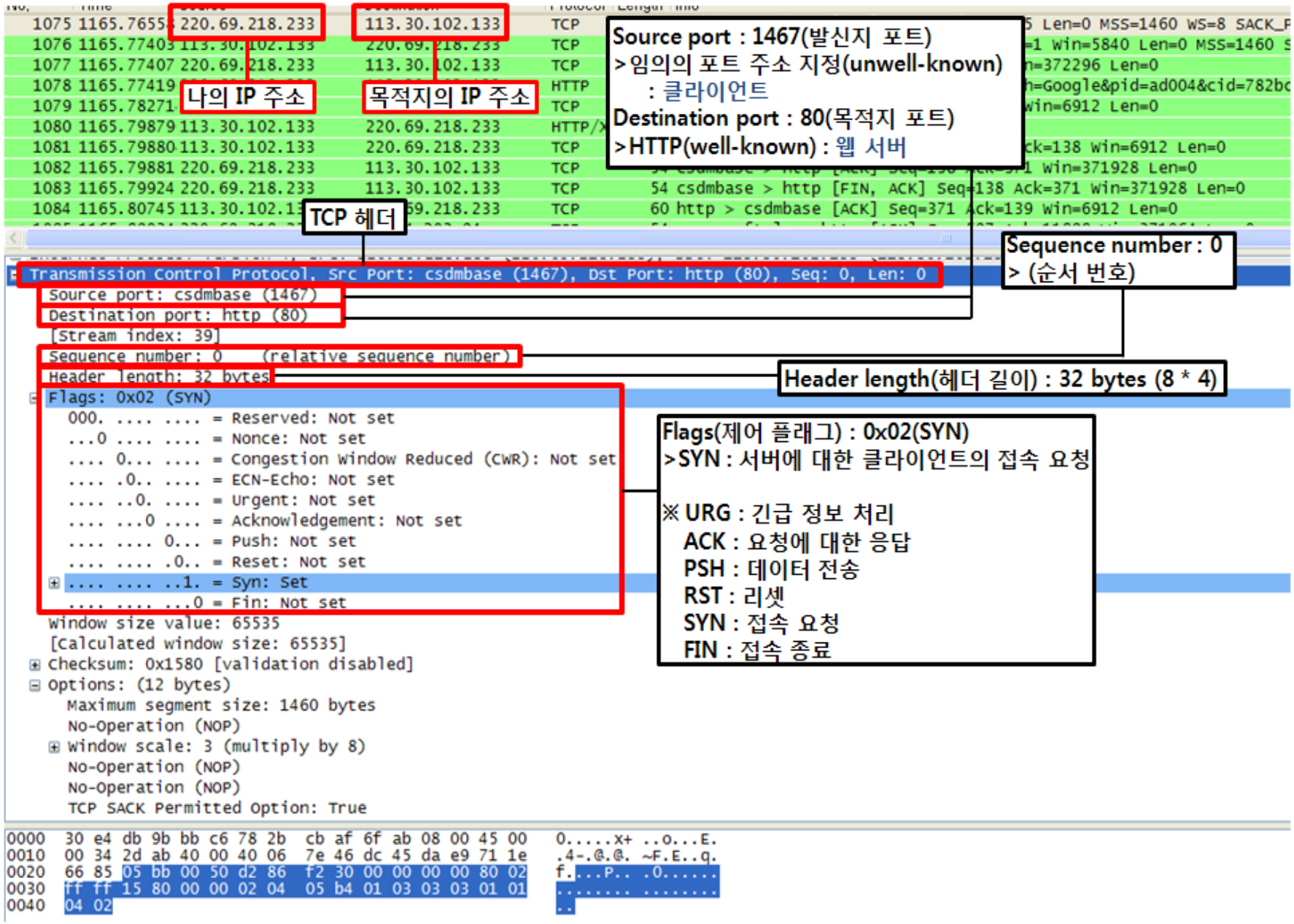


⑨ 프레임 속의 캡슐화

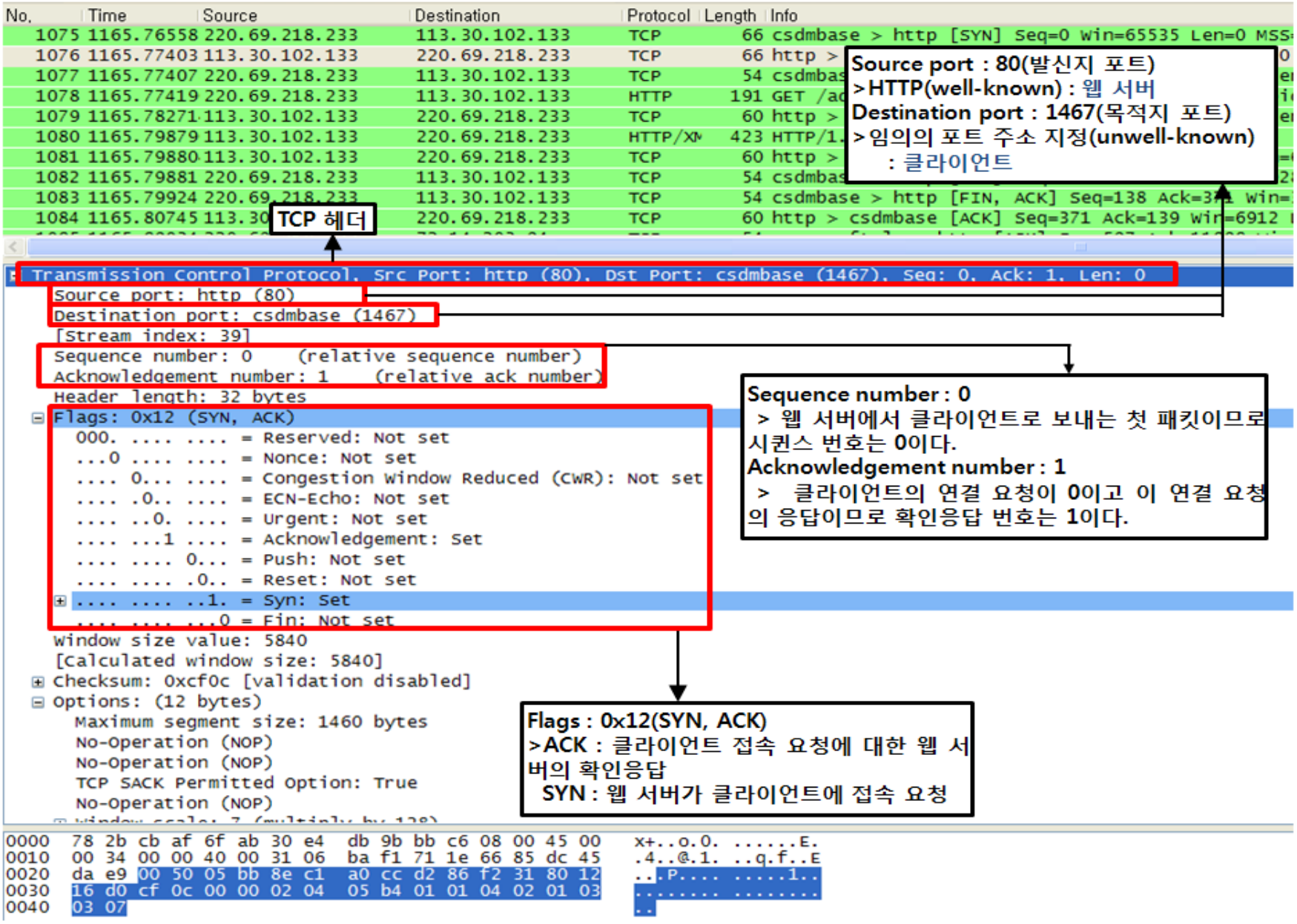


- 프레임 속의 캡슐화 -

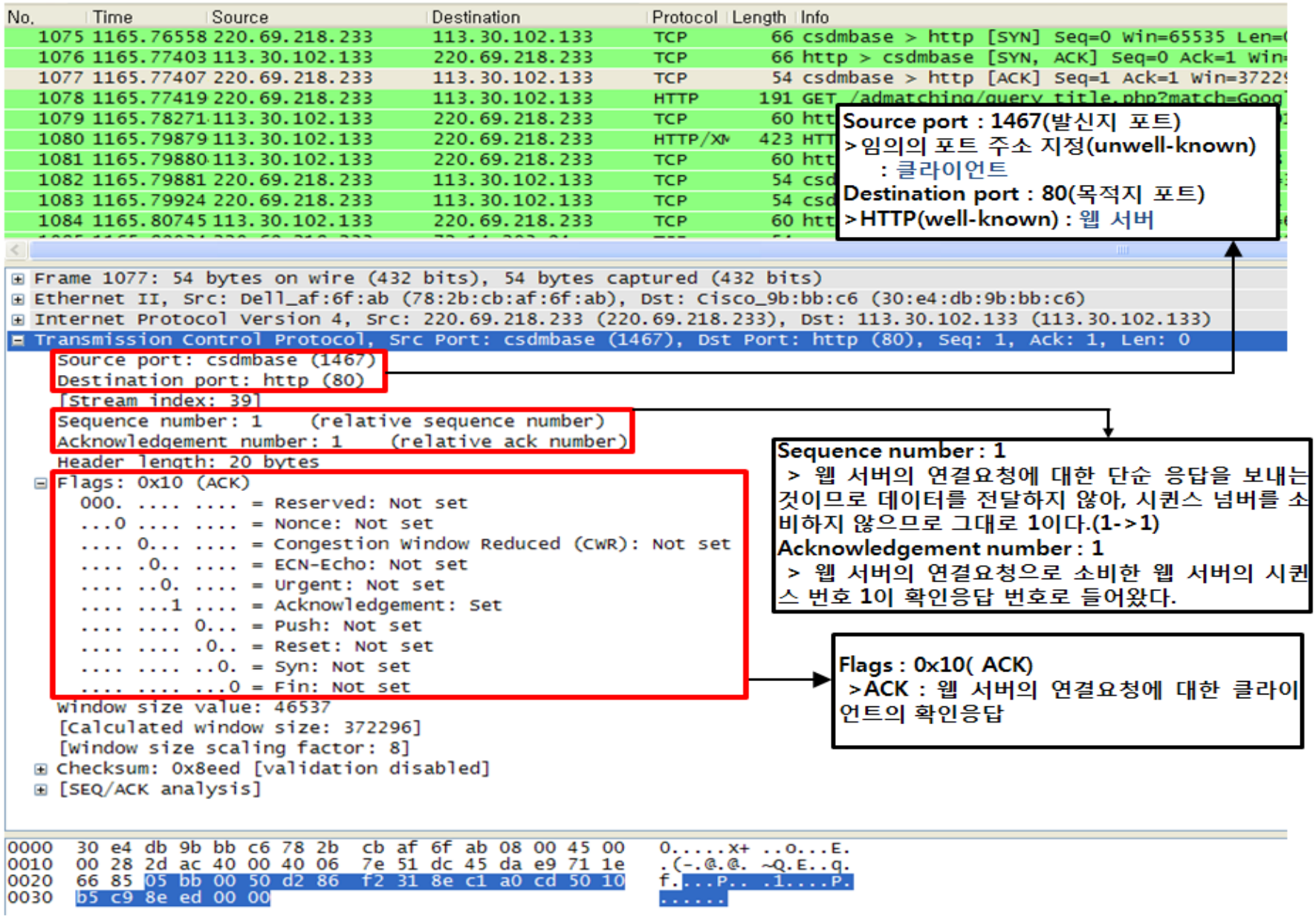
⑩(1) TCP 패킷 분석 (인터넷 접속 후 클라이언트가 웹 서버에 접속 연결 요청(SYN))



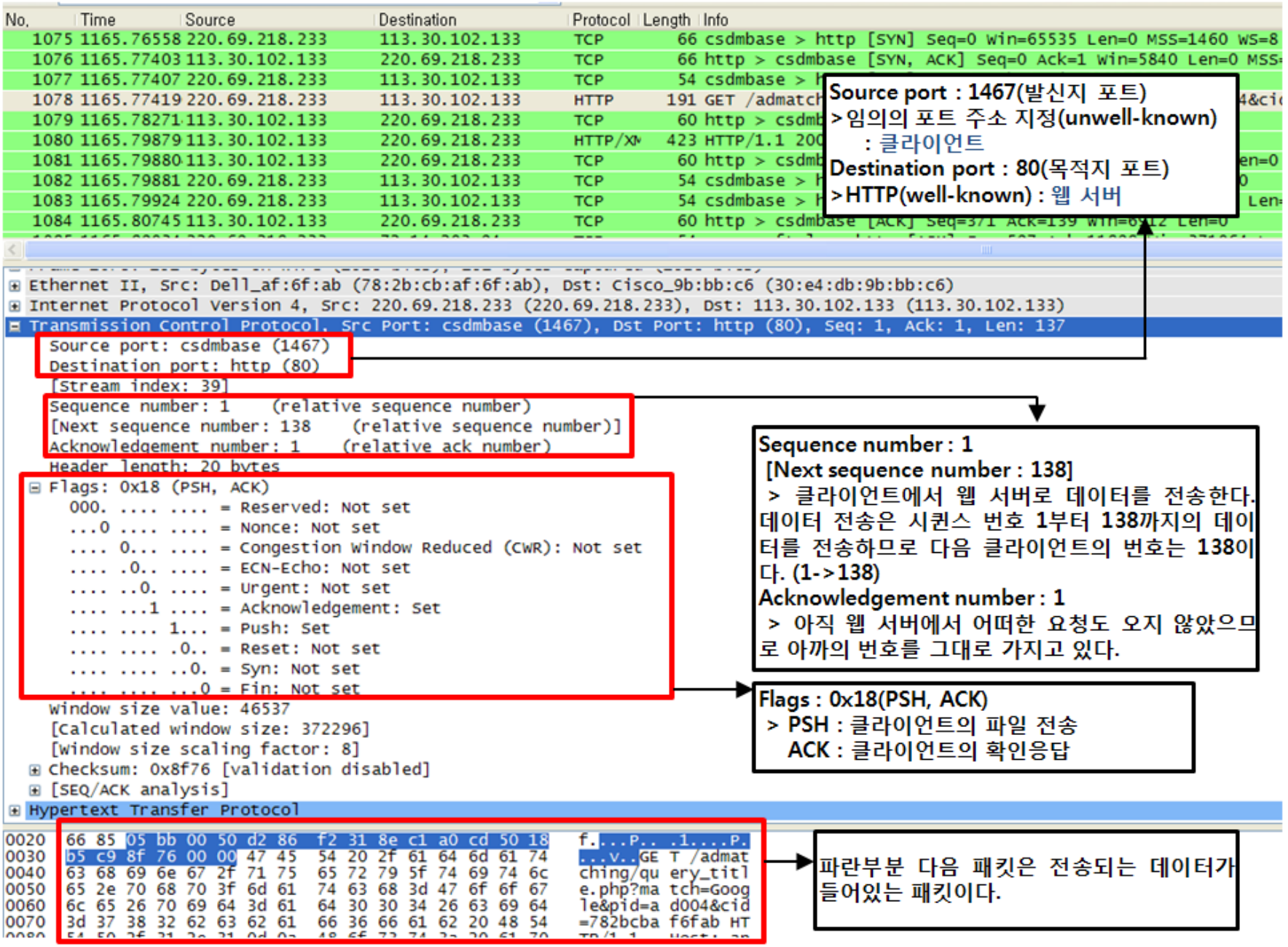
⑩(2) TCP 패킷 분석 (웹 서버가 클라이언트의 접속연결 요청에 대한 응답(ACK)과 웹 서버의 접속 연결 요청(SYN))



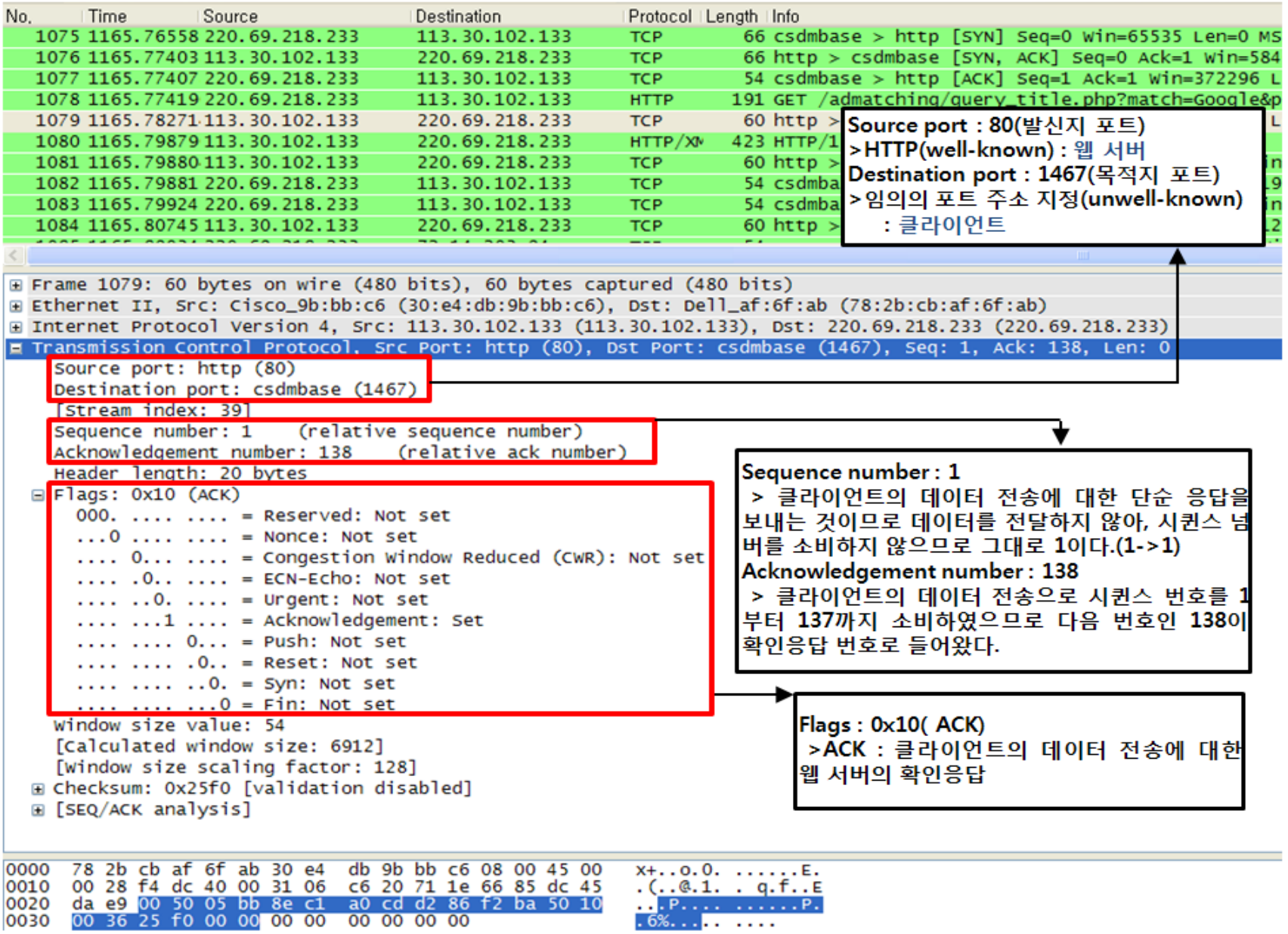
⑩(3) TCP 패킷 분석(웹 서버의 연결요청에 대한 클라이언트의 확인응답(ACK))



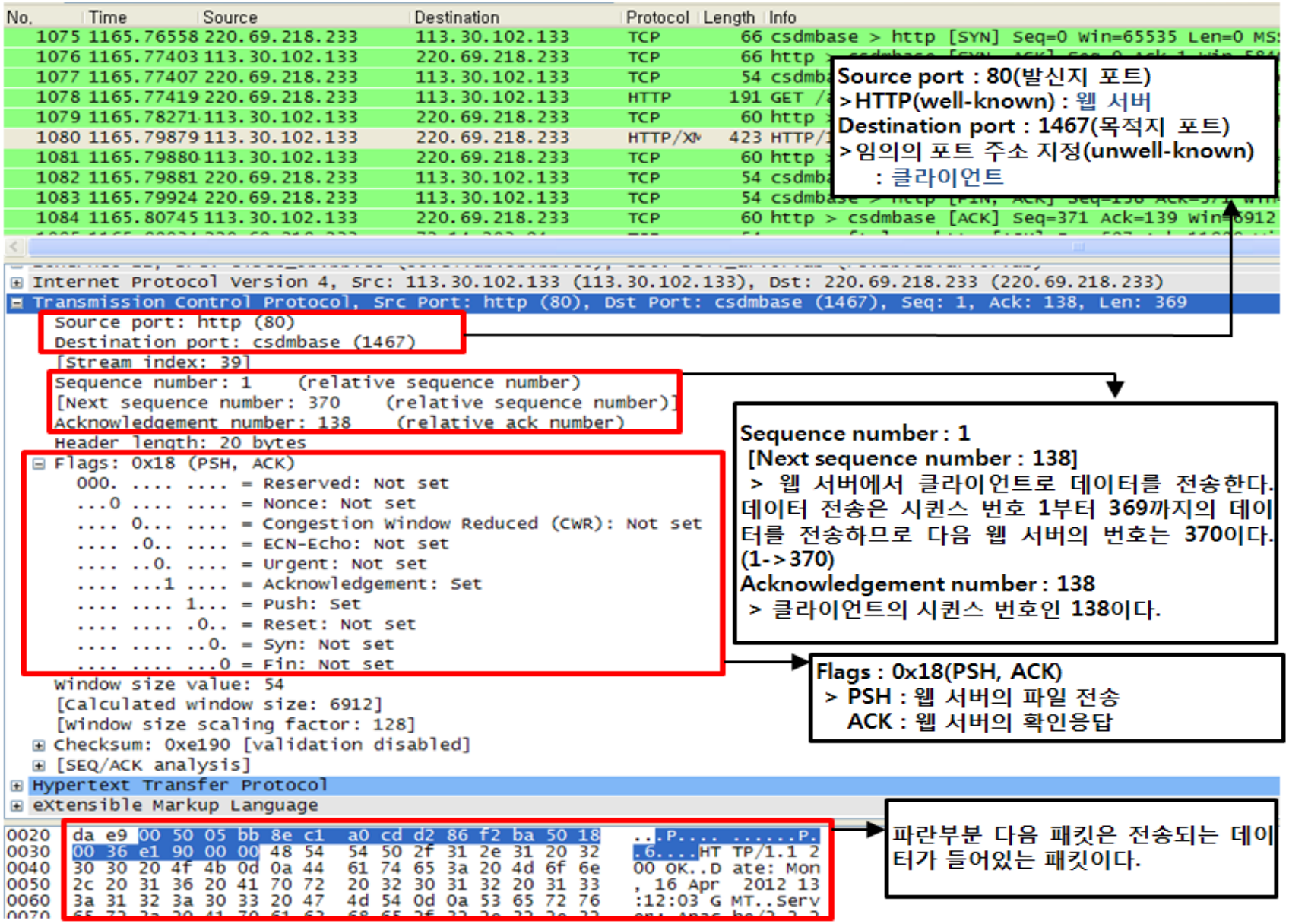
⑩(4) TCP 패킷 분석(웹 서버에 클라이언트의 데이터 전송(PSH)과 확인응답(ACK))



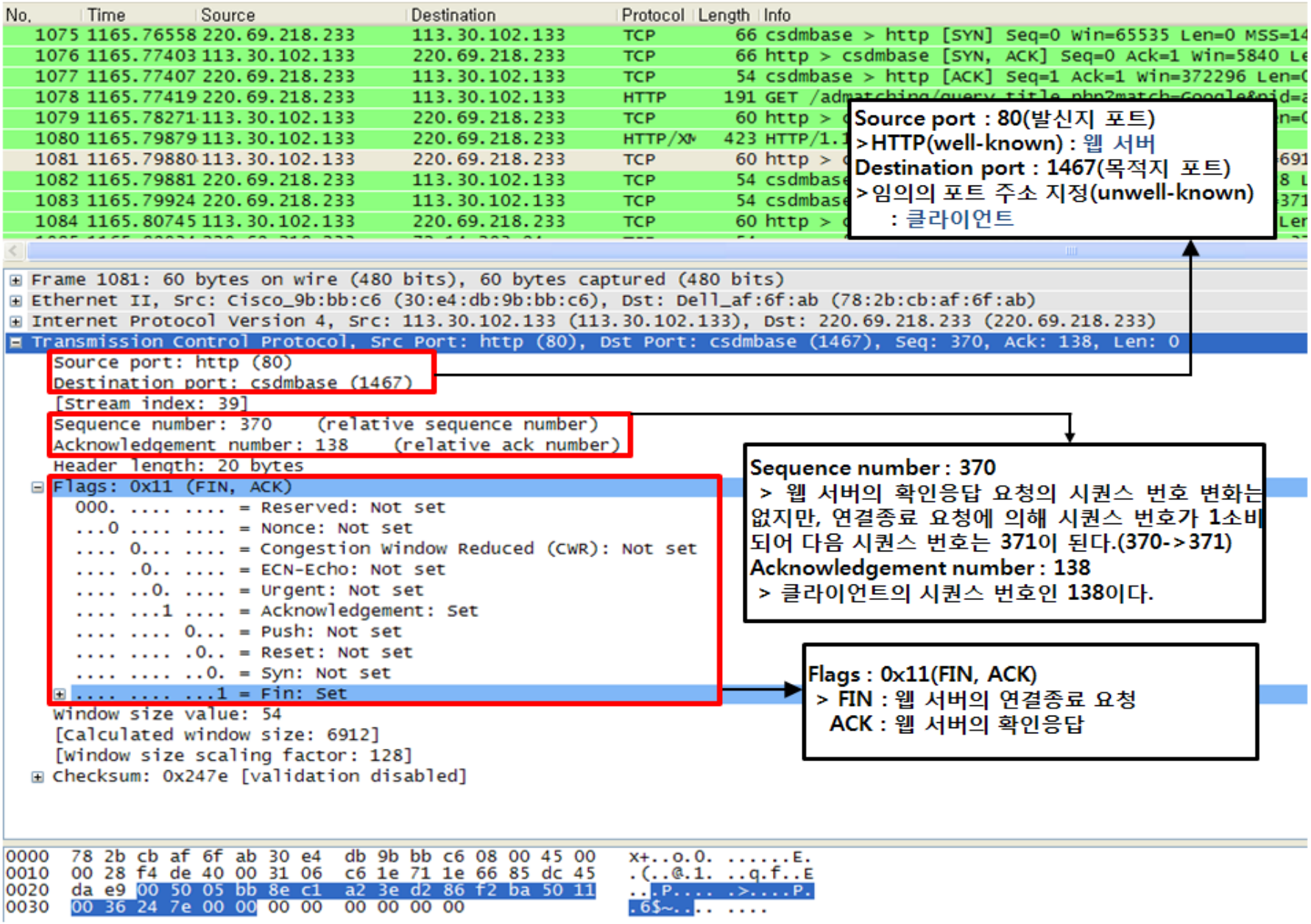
⑩(5) TCP 패킷 분석(클라이언트의 데이터 전송에 대한 웹 서버의 확인응답(ACK))



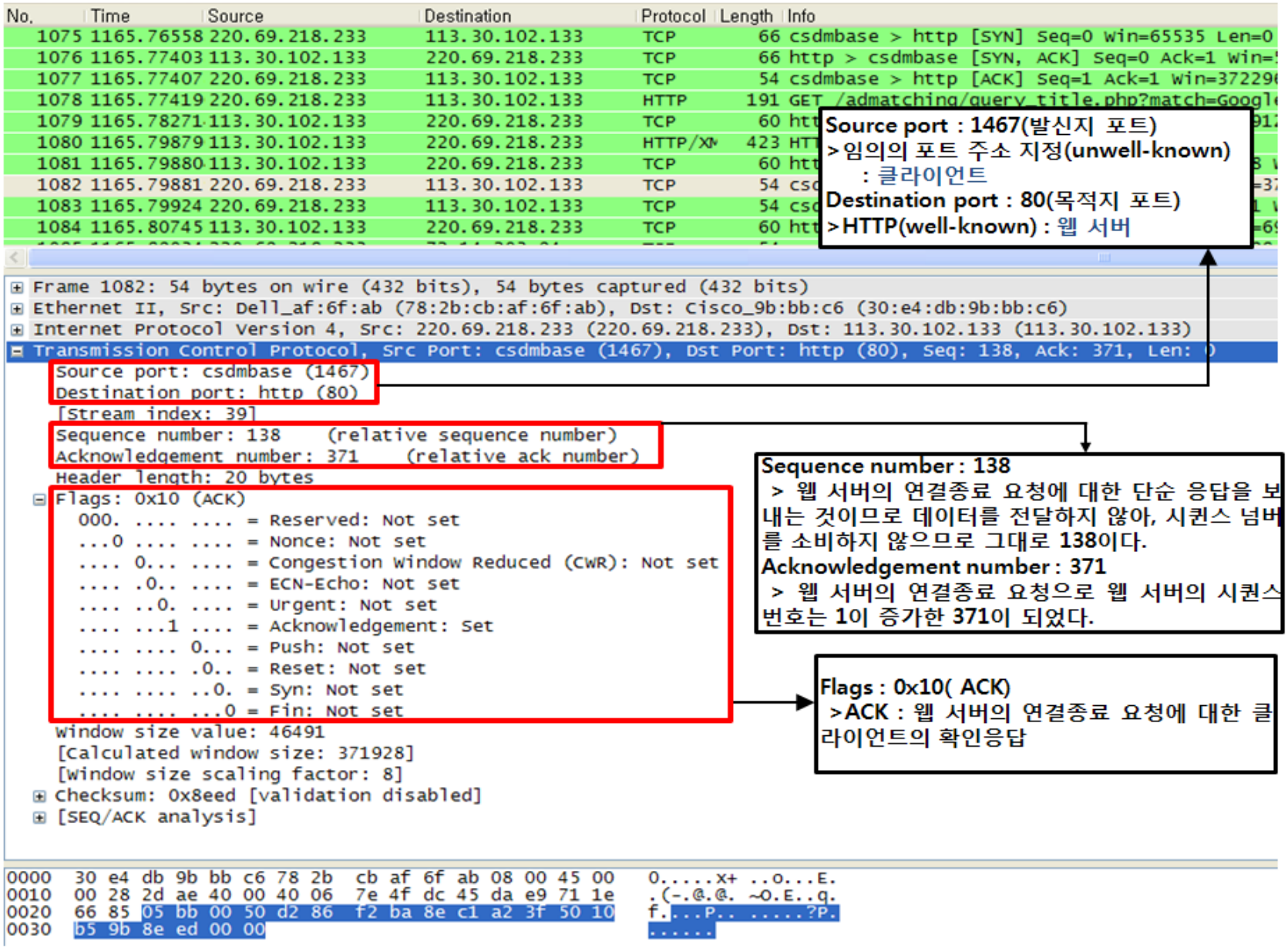
⑩(6) TCP 패킷 분석(클라이언트에 웹 서버의 데이터 전송(PSH)과 클라이언트의 데이터 전송에 대한 확인응답(ACK))



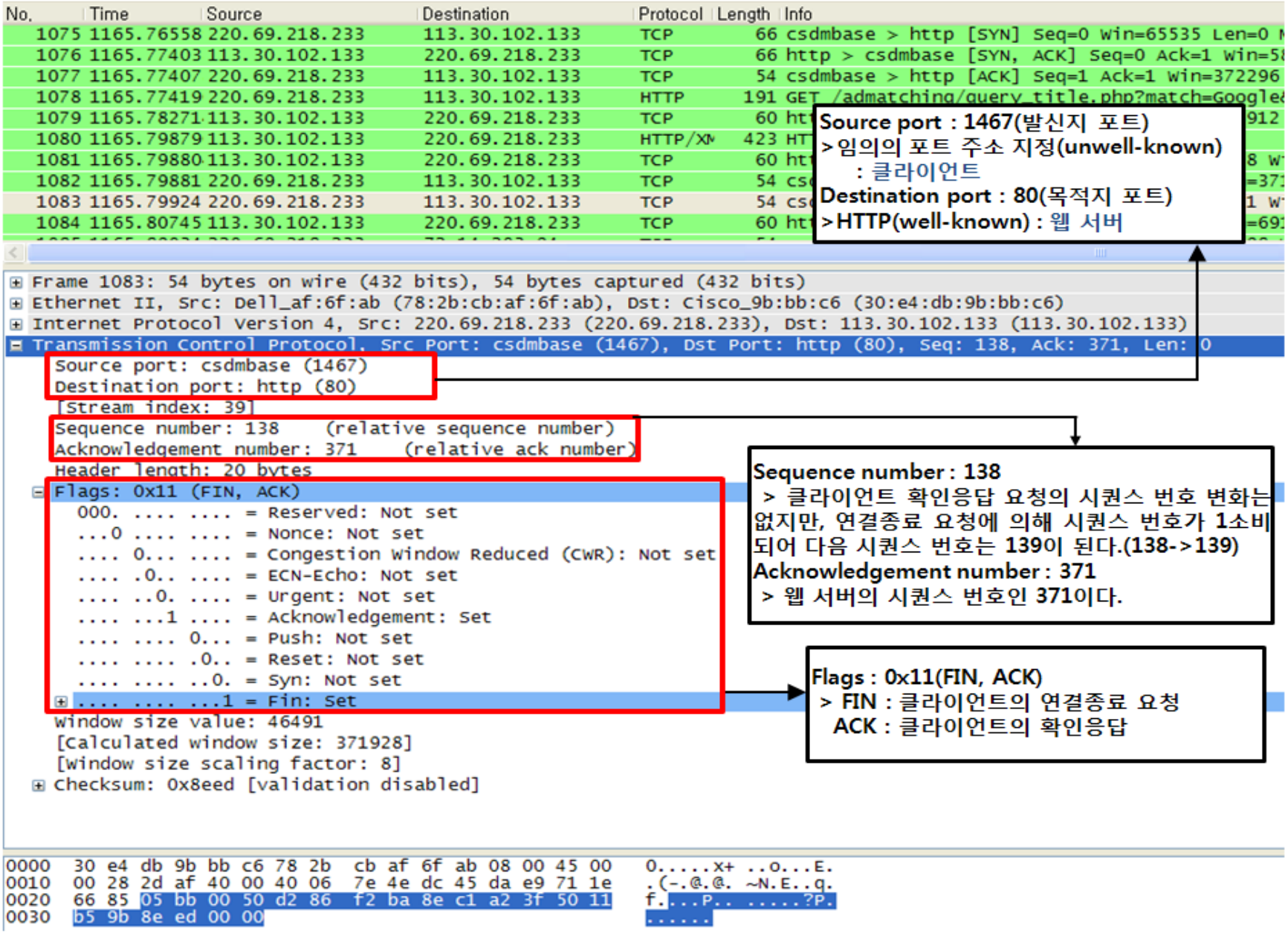
⑩(7) TCP 패킷 분석(웹 서버의 연결종료 요청(FIN)과 확인응답(ACK))



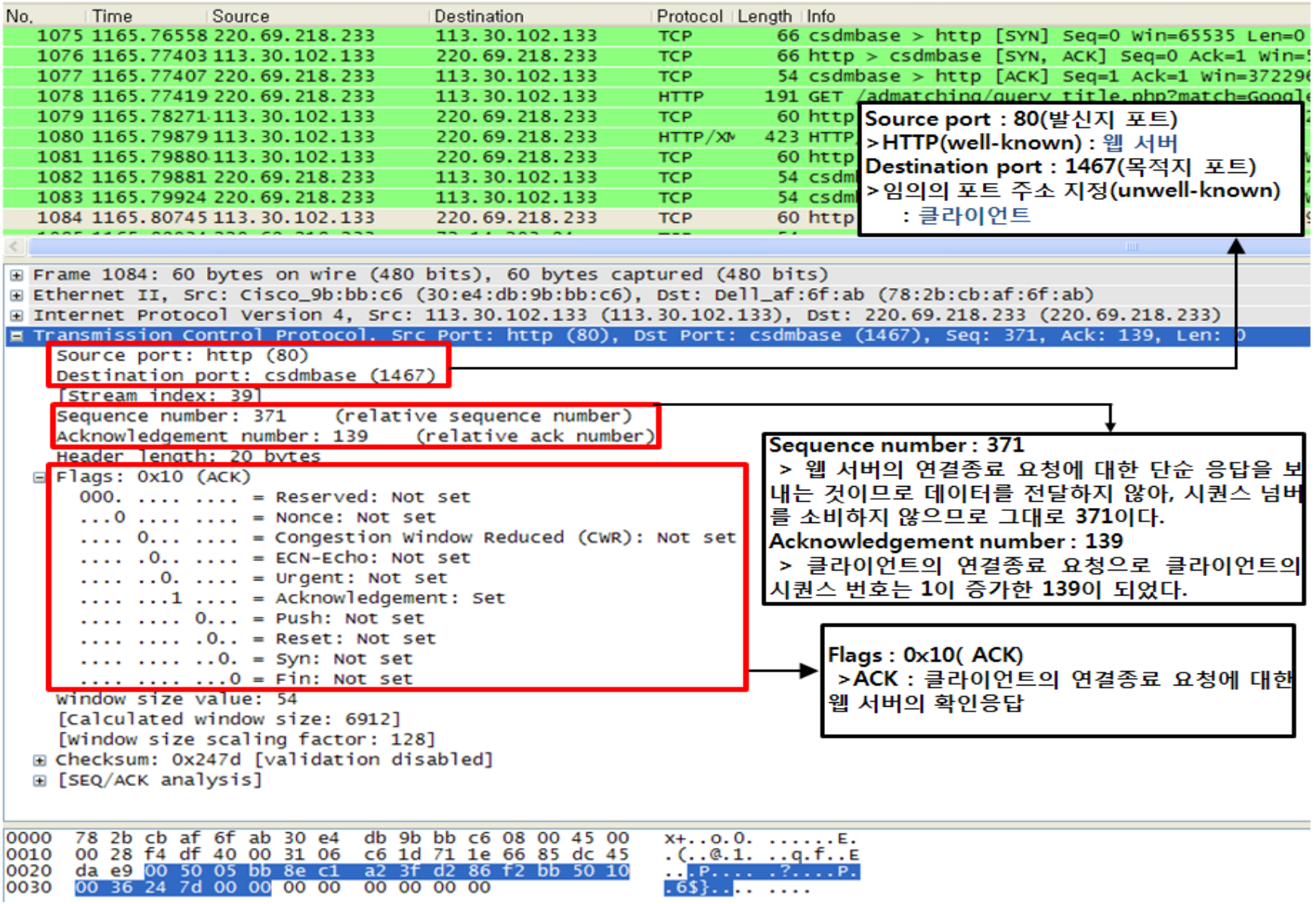
⑩(8) TCP 패킷 분석
(웹 서버의 연결종료 요청에 대한 클라이언트의 확인응답(ACK))



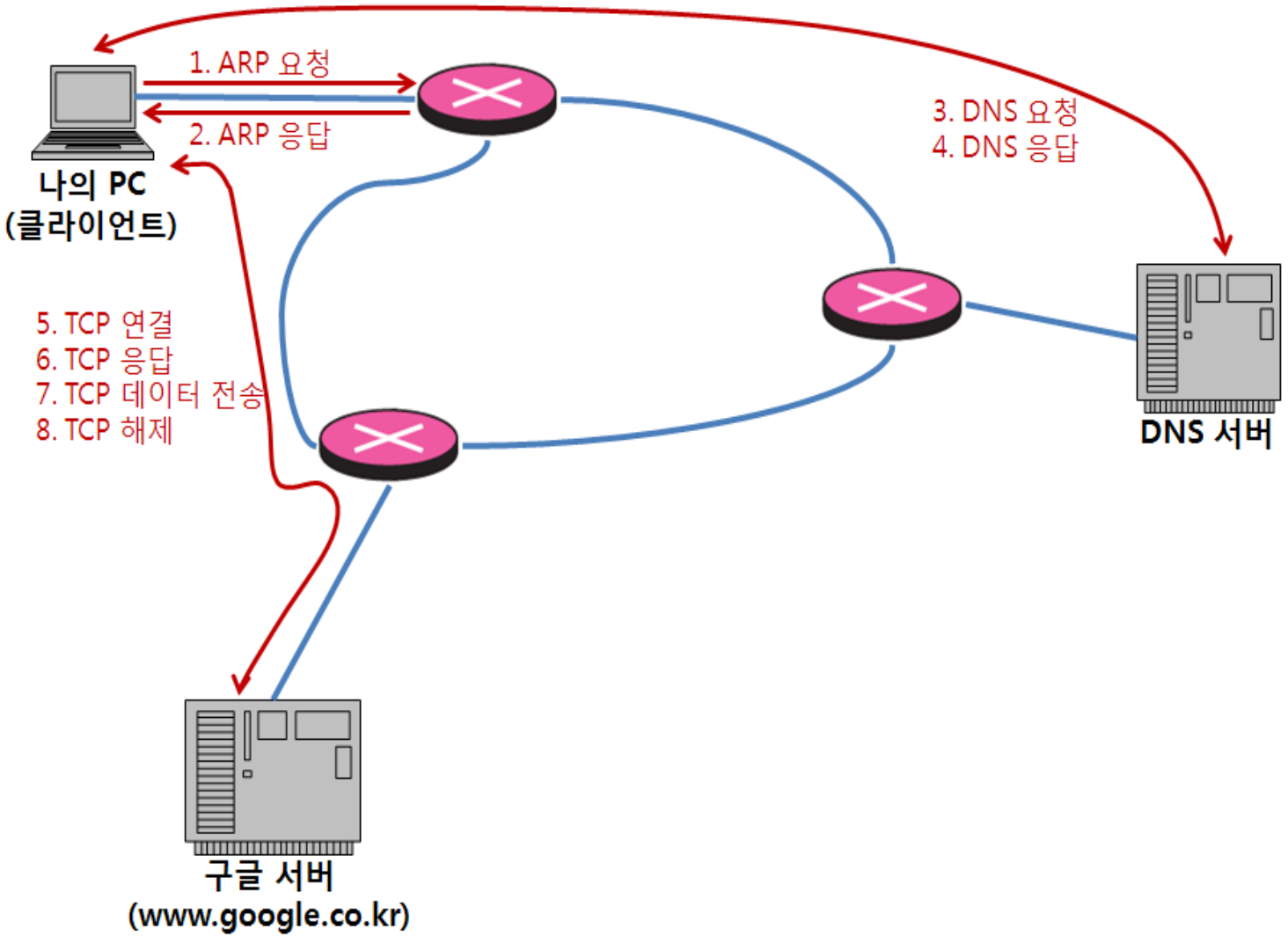
⑩(9) TCP 패킷 분석(클라이언트의 연결종료 요청(FIN)과 확인응답(ACK))



⑩(10) TCP 패킷 분석
(클라이언트의 연결종료 요청에 대한 웹 서버의 확인응답(ACK))



⑪ 결론


1. ARP 요청
2. ARP 응답
3. DNS 요청
4. DNS 응답
나의 PC
(클라이언트)
5. TCP 연결
6. TCP 응답
7. TCP 데이터 전송
8. TCP 해제
DNS 서버
구글 서버
(www.google.co.kr)