2026/09/08, 2026 네트워크 기초 세미나

TCP/IP 프로토콜

- 8장 ARP, 9장 ICMPv4, 10장 모바일IP -

김 민 식 (minsik@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

- ARP(Address Resolution Protocol)
- ICMPv4(Internet Control Message Protocol Version 4)
- 모바일 IP(Mobile IP)

목 차

- ARP(Address Resolution Protocol)
- ICMPv4(Internet Control Message Protocol Version 4)
- 모바일 IP(Mobile IP)

ARP

- 주소의 종류
  - 논리 주소(Logical Address)
    - 네트워크 계층에서 장치의 출발지와 목적지를 식별하여 통신하기 위해 사용하는 주소
    - 네트워크를 연결하기 위해 사용되는 모든 프로토콜들은 논리 주소를 필요로 함
    - TCP/IP에서 논리 주소로 IP 주소를 사용하며, IPv4는 32비트, IPv6는 128비트의 길이를 가짐
  - 로컬 주소(Local Address)
    - 데이터 링크 계층에서 동일한 로컬 네트워크 내의 장치를 식별하고 통신하기 위해 사용하는 주소
    - 하드웨어 내에 구현되므로 물리 주소라고 불림
    - 이더넷이나 토큰링의 물리주소는 호스트나 라우터 내에 설치된 NIC에 들어있으며 48비트의 길이를 가짐

*NIC(Network Interface Card):
네트워크에 연결할 수 있도록 회로를 포함한 장치 내부 또는 외부의 전자 장치

ARP

- 주소 변환
  - 필요성
    - 서로 다른 계층에서 사용하는 논리 주소(IP, Internet Protocol 주소)와 로컬 주소(MAC, Media Access Control 주소)를 연결하여 실제 데이터 전송을 가능하게 하기 위함
  - 종류
    - 정적 변환(Static Mapping)
    - 동적 변환(Dynamic Mapping)

ARP

- 주소 변환 – 종류 (1/2)

*NIC(Network Interface Card):
네트워크에 연결할 수 있도록 회로를
포함한 장치 내부 또는 외부의 전자 장치

  - 정적 변환
    - 관리자가 논리 주소와 로컬 주소의 대응 관계를 변환표에 직접 설정하여 사용하는 주소 변환 방식
    - 한계점
      1. 기계는 NIC를 바꿀 수 있고 결과적으로 새 물리 주소를 갖게 됨
      2. LocalTalk와 같은 LAN에서는 컴퓨터가 켜질 때마다 물리 주소가 변함
      3. 이동 컴퓨터는 하나의 물리적 네트워크에서 다른 네트워크로 이동할 수 있고 이러한 경우 물리 주소는 변하게 됨
      - 한계를 극복하기 위해 정적 변환 테이블은 주기적으로 갱신되어야 함

ARP

- 주소 변환 – 종류 (2/2)
  - 동적 변환
    - ARP 요청과 응답을 통해 논리 주소에 대응하는 로컬 주소를 자동으로 확인하여 변환표에 등록하는 방식
    - 한계점
      1. ARP 캐시에 정보가 없거나 만료될 때 ARP 요청을 브로드캐스트하므로 네트워크 트래픽 발생
      2. 최초 통신 시 물리 주소를 찾는 과정에서 전송 지연 발생
      3. 인증 절차가 없어 위조된 ARP 응답을 보내는 ARP 스푸핑 공격에 취약

ARP

- 정의
  - 동일한 로컬 네트워크에서 목적지의 IPv4 주소에 대응하는 MAC 주소를 알아내는 주소 변환 프로토콜
- 특징
  - IPv4 주소를 MAC 주소로 변환
  - 동일한 로컬 네트워크 내에서 동작
  - ARP 요청은 브로드캐스트 방식으로 전송
  - ARP 응답은 일반적으로 유니캐스트 방식으로 전송
- 필요성
  - IP 주소만으로는 로컬 네트워크에서 프레임을 전달할 수 없음
  - 목적지 IP 주소에 대응하는 MAC 주소를 확인하기 위해 필요함

ARP

- ARP가 사용되는 경우
  1. 같은 네트워크의 다른 호스트로 패킷을 전송할 때
     - 목적지 호스트의 IP 주소를 물리 주소(MAC 주소)로 변환함
  2. 다른 네트워크의 호스트로 패킷을 전송할 때
     - 라우팅 테이블에서 찾은 다음 홉 또는 디폴트 라우터의 IP 주소를 물리 주소(MAC 주소)로 변환함
  3. 라우터가 다른 네트워크로 전달할 패킷을 수신했을 때
     - 다음 홉 라우터의 IP 주소를 물리 주소(MAC 주소)로 변환함
  4. 라우터가 같은 네트워크의 호스트로 전달할 패킷을 수신했을 때
     - 목적지 호스트의 IP 주소를 물리 주소(MAC 주소)로 변환함

ARP

- 패킷 구조
  - 하드웨어 유형(Hardware Type)
    - ARP가 수행되고 있는 네트워크 유형을 정의하는 16비트 필드
      - e.g., 이더넷의 경우, 1의 값을 가짐
  - 프로토콜 유형(Protocol Type)
    - 프로토콜을 정의하는 16비트 필드
      - e.g., IPv4 경우, 0800_{16}의 값을 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td>Protocol Type</td></tr>
<tr><td>Hardware length</td><td>Protocol length</td><td>Operation
Request 1, Reply 2</td></tr>
<tr><td colspan="3">Sender hardware address
(For example, 6 bytes for Ethernet)</td></tr>
<tr><td colspan="3">Sender protocol address
(For example, 4 bytes for IP)</td></tr>
<tr><td colspan="3">Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled in a request)</td></tr>
<tr><td colspan="3">Target protocol address
(For example, 4 bytes for IP)</td></tr>
</table>

ARP

- 패킷 구조
  - 하드웨어 길이(Hardware Length)
    - 물리 주소의 길이를 바이트 단위로 정의하는 8비트 필드
      - e.g., 이더넷의 경우, 6의 값을 가짐
  - 프로토콜 길이(Protocol Length)
    - 논리 주소의 길이를 바이트 단위로 정의하는 8비트 필드
      - e.g., IPv4 경우, 4의 값을 가짐
  - 동작(Operation)
    - 패킷의 유형을 정의하는 16비트 필드
      - ARP 요청(1)과 ARP 응답(2)의 두 가지 유형이 정의되어 있음

<table>
<tr><td colspan="2">Hardware Type</td><td>Protocol Type</td></tr>
<tr><td>Hardware length</td><td>Protocol length</td><td>Operation
Request 1, Reply 2</td></tr>
<tr><td colspan="3">Sender hardware address
(For example, 6 bytes for Ethernet)</td></tr>
<tr><td colspan="3">Sender protocol address
(For example, 4 bytes for IP)</td></tr>
<tr><td colspan="3">Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled in a request)</td></tr>
<tr><td colspan="3">Target protocol address
(For example, 4 bytes for IP)</td></tr>
</table>

ARP

- 패킷 구조
  - 발신지 하드웨어 주소(Sender Hardware Address)
    - 송신자의 물리 주소를 나타내는 가변 길이의 필드
      - e.g., 이더넷의 경우, 6바이트의 크기를 가짐
  - 발신지 프로토콜 주소(Sender Protocol Address) 필드
    - 송신자의 논리 주소를 나타내는 가변 길이의 필드
      - e.g., IP 프로토콜의 경,우 4바이트의 크기를 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td>Protocol Type</td></tr>
<tr><td>Hardware length</td><td>Protocol length</td><td>Operation
Request 1, Reply 2</td></tr>
<tr><td colspan="3">Sender hardware address
(For example, 6 bytes for Ethernet)</td></tr>
<tr><td colspan="3">Sender protocol address
(For example, 4 bytes for IP)</td></tr>
<tr><td colspan="3">Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled in a request)</td></tr>
<tr><td colspan="3">Target protocol address
(For example, 4 bytes for IP)</td></tr>
</table>

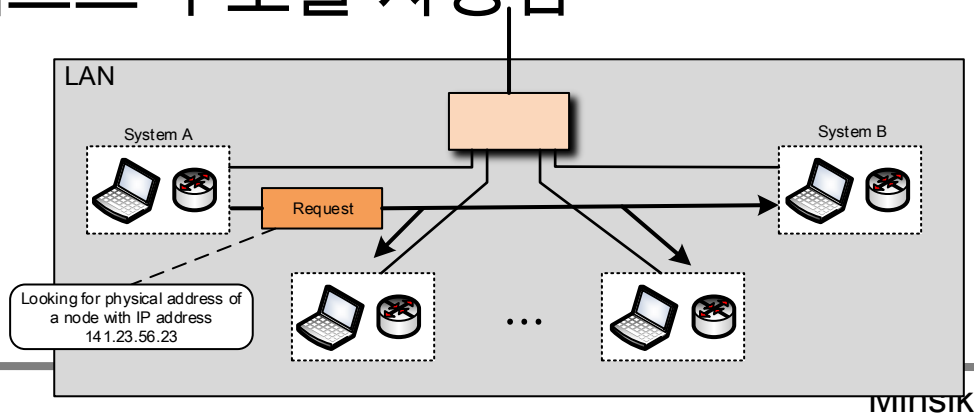
ARP

- 패킷 구조
  - 타깃 하드웨어 주소(Target Hardware Address)
    - 타깃의 물리 주소를 나타내는 가변 길이의 필드
      - e.g., 이더넷의 경우, 6바이트의 크기를 가짐
      - e.g., ARP 요청 메시지의 경우, 송신자는 타깃의 물리 주소를 알지 못하므로 0의 값을 가짐
  - 타깃 프로토콜 주소(Target Protocol Address)
    - 타깃의 논리 주소를 나타내는 가변 길이의 필드
      - e.g., IPv4의 경우, 4바이트의 크기를 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td>Protocol Type</td></tr>
<tr><td>Hardware length</td><td>Protocol length</td><td>Operation
Request 1, Reply 2</td></tr>
<tr><td colspan="3">Sender hardware address
(For example, 6 bytes for Ethernet)</td></tr>
<tr><td colspan="3">Sender protocol address
(For example, 4 bytes for IP)</td></tr>
<tr><td colspan="3">Target hardware address
(For example, 6 bytes for Ethernet)
(It is not filled in a request)</td></tr>
<tr><td colspan="3">Target protocol address
(For example, 4 bytes for IP)</td></tr>
</table>

ARP

- 동작 과정 – 요청
  1. 송신자는 수신자의 IP 주소를 알고 있음
  2. 송신자는 수신자의 MAC 주소를 알아내기 위해 ARP 요청 메시지를 생성함
     - 송신자의 물리 주소와 IP 주소, 수신자의 IP 주소는 채워지지만 타깃의 물리 주소 필드는 0으로 채워짐
  3. ARP 요청은 데이터 링크 계층에서 프레임으로 캡슐화되어 브로드캐스트로 전송됨
     - 출발지 MAC 주소는 송신자의 MAC 주소, 목적지 MAC 주소는 브로드캐스트 주소를 사용함

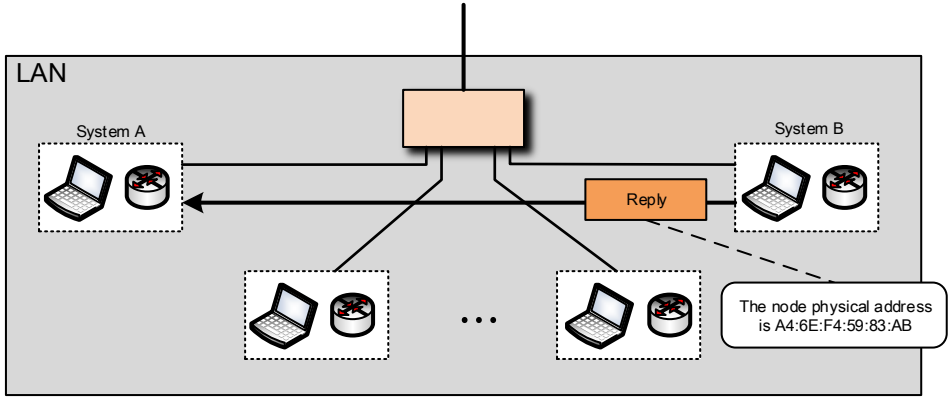


a. ARP request is broadcast

ARP

- 동작 과정 – 응답

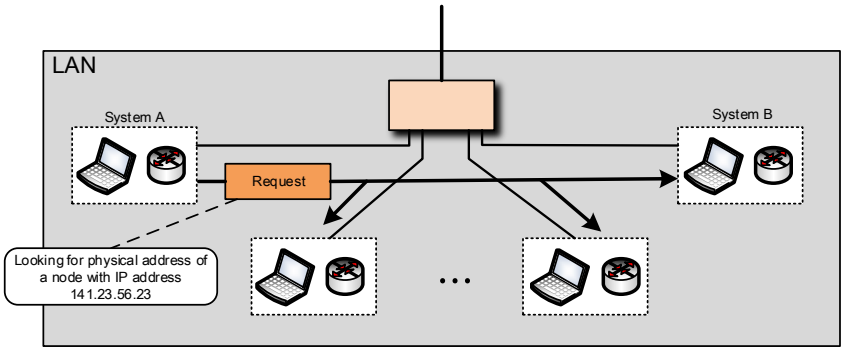
4. 수신자는 브로드캐스트된 ARP 요청을 수신하고, 요청된 IP 주소가 자신의 IP 주소와 일치하는지 확인함
   - 일치하지 않는 장치는 해당 ARP 요청을 폐기함
5. 수신자는 자신의 MAC 주소를 포함한 ARP 응답을 송신자에게 유니캐스트로 전송함
6. 송신자는 ARP 응답을 통해 수신자의 MAC 주소를 확인함
7. 송신자는 IP 데이터그램을 수신자의 MAC 주소를 목적지로 하는 프레임에 캡슐화하여 유니캐스트로 전송함



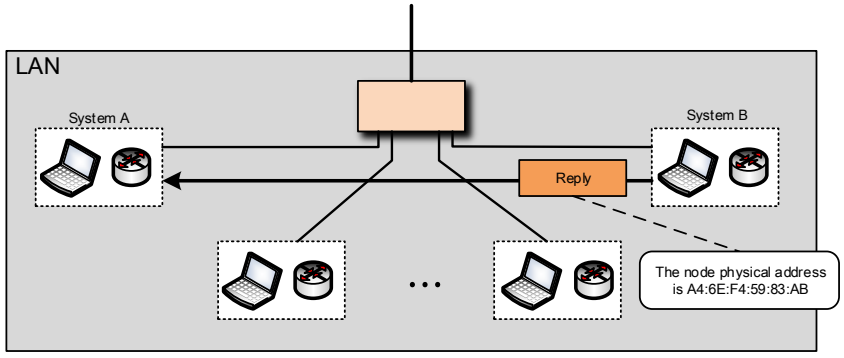
a. ARP reply is unicast

ARP

- 주소 매핑 과정
  1. 송신자는 목적지 IP 주소에 대응하는 MAC 주소가 ARP 캐시에 있는지 확인함
  2. 정보가 없으면 목적지 IP 주소를 포함한 ARP Request를 브로드캐스트함
  3. 해당 IP 주소를 가진 수신자는 자신의 MAC 주소를 ARP Reply로 응답함
  4. 송신자는 확인한 목적지 IP 주소와 MAC 주소의 대응 관계를 ARP 캐시에 저장하여 이후 데이터 전송에 사용함



a. ARP request is broadcast



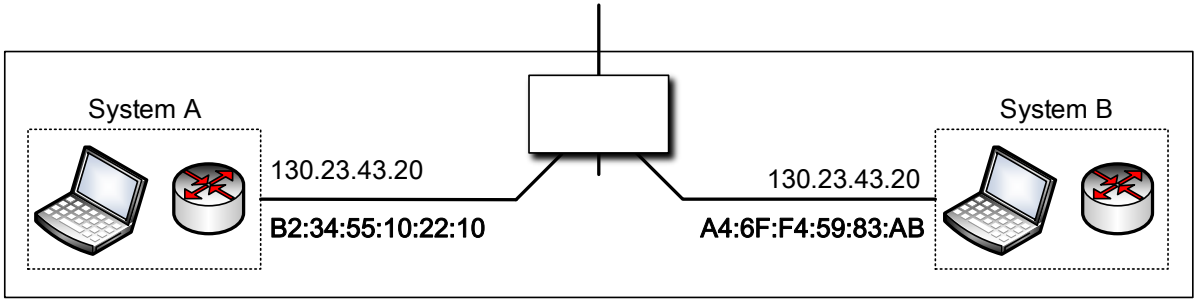
a. ARP reply is unicast

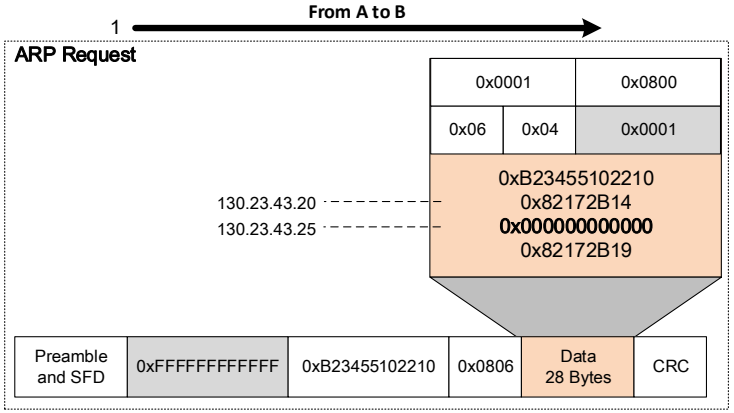
ARP

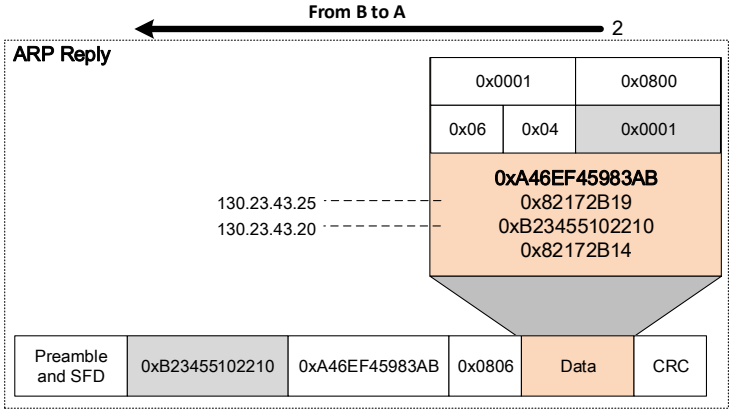
- 예제 8.1

아래와 같은 주소들을 가진 호스트 A와 B가 있다. 같은 이더넷 네트워크 상에 있는 두 호스트 간에, 호스트 A가 호스트 B에게 패킷을 보낼 때 ARP 요청과 응답 패킷을 보여라.

- 풀이
  - 호스트 A
    - (IP 주소) 130.23.43.20
    - (MAC 주소) B2:34:55:10:22:10
  - 호스트 B
    - (IP 주소) 130.23.43.25
    - (MAC 주소) A4:6E:F4:59:83:AB

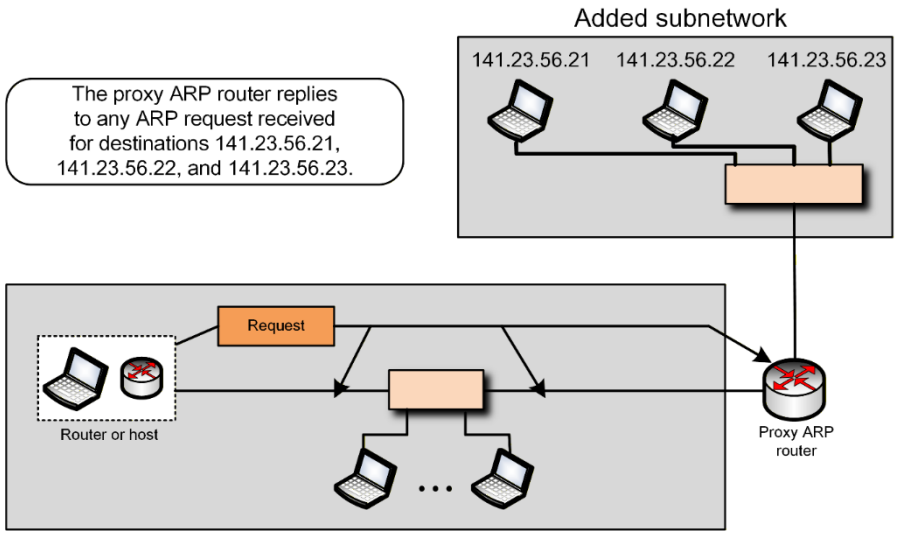






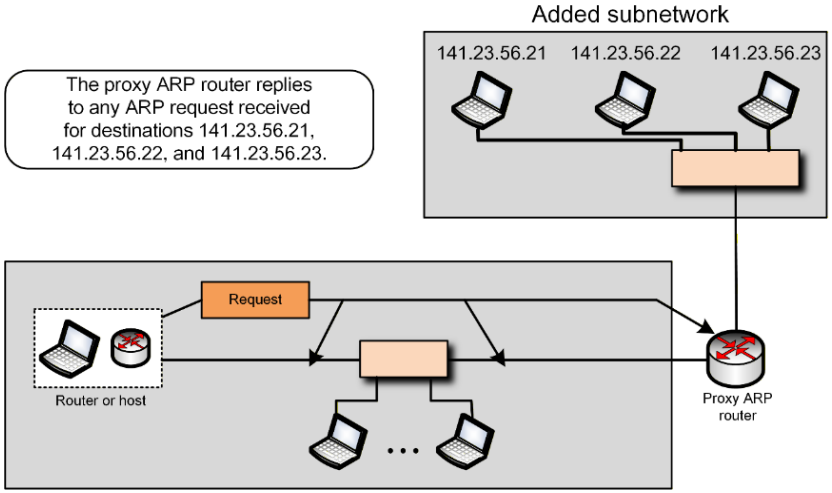
ARP

- 프록시 ARP (1/2)
  - 정의
    - 라우터가 다른 네트워크에 있는 장치를 대신하여 ARP 요청에 자신의 MAC 주소로 응답하는 방식
  - 목적
    - 다른 네트워크의 호스트를 같은 로컬 네트워크에 있는 것처럼 인식하게 하여, 호스트의 설정을 변경하지 않고 통신할 수 있도록 하기 위함



ARP

- 프록시 ARP (2/2)
  - 동작
    1. 송신자가 다른 네트워크의 호스트에 대한 ARP 요청을 브로드캐스트하면 프록시 ARP 라우터가 이를 수신함
    2. 프록시 ARP 라우터는 대상 호스트를 대신하여 자신의 MAC 주소를 ARP 응답으로 송신자에게 전달함
    3. 송신자가 해당 MAC 주소로 IP 패킷을 전송하면 프록시 ARP 라우터가 실제 목적지 호스트로 패킷을 전달함



ARP

- ATMARP (1/7)
  - 정의
    - ATM 네트워크에서 목적지의 IP 주소를 ATM 주소로 변환하여 가상 연결을 설정하는 주소 변환 프로토콜
  - 이더넷 LAN과 ATM 네트워크 비교
    - 이더넷 LAN
      - 이더넷 LAN에서는 브로드캐스트 방식으로 ARP 요청을 동일한 로컬 네트워크에 전달함
    - ATM 네트워크
      - ATM 네트워크에서는 ARP 요청을 브로드캐스트 방식으로 전달할 수 없어 별도의 주소 변환 방식이 필요함

ARP

- ATMARP (2/7)
  - 패킷 구조
    - 하드웨어 유형(Hardware Type) 필드
      - 물리적 네트워크를 정의하는 16비트 필드
      - ATM 네트워크의 경우, 0013_{16}의 값을 가짐
    - 프로토콜 유형(Protocol Type) 필드
      - 프로토콜 유형을 정의하는 16비트 필드
      - IPv4 프로토콜의 경우, 0800_{16}의 값을 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td colspan="2">Protocol Type</td></tr>
<tr><td>Sender Hardware Length</td><td>Reserved</td><td colspan="2">Operation</td></tr>
<tr><td>Sender Protocol Length</td><td>Target Hardware Length</td><td>Reserved</td><td>Target Protocol Length</td></tr>
<tr><td colspan="4">Sender hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Sender Protocol address</td></tr>
<tr><td colspan="4">Target hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Target protocol address</td></tr>
</table>

ARP

- ATMARP (3/7)
  - 패킷 구조
    - 송신자 하드웨어 길이(Sender Hardware Length) 필드
      - 송신자의 물리 주소의 길이를 바이트 단위로 정의하는 8비트 필드
      - ATM 네트워크의 경우, 20의 값을 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td colspan="2">Protocol Type</td></tr>
<tr><td>Sender Hardware Length</td><td>Reserved</td><td colspan="2">Operation</td></tr>
<tr><td>Sender Protocol Length</td><td>Target Hardware Length</td><td>Reserved</td><td>Target Protocol Length</td></tr>
<tr><td colspan="4">Sender hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Sender Protocol address</td></tr>
<tr><td colspan="4">Target hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Target protocol address</td></tr>
</table>

ARP

- ATMARP (4/7)
  - 패킷 구조
    - 동작(Operation) 필드
      - 패킷의 유형을 정의하는 16비트 필드
      - 다섯 개의 패킷 유형이 정의됨

Message	OPER value
Request	1
Reply	2
Inverse Request	8
Inverse Reply	9
NACK	10

    - 송신자 프로토콜 길이(Sender Protocol Length) 필드
      - 논리 주소의 길이를 바이트 단위로 정의한 8비트 필드
      - IPv4의 경우, 4의 값을 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td colspan="2">Protocol Type</td></tr>
<tr><td>Sender Hardware Length</td><td>Reserved</td><td colspan="2">Operation</td></tr>
<tr><td>Sender Protocol Length</td><td>Target Hardware Length</td><td>Reserved</td><td>Target Protocol Length</td></tr>
<tr><td colspan="4">Sender hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Sender Protocol address</td></tr>
<tr><td colspan="4">Target hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Target protocol address</td></tr>
</table>

ARP

- ATMARP (5/7)
  - 패킷 구조
    - 타깃 하드웨어 길이(Target Hardware Length) 필드
      - 수신자의 물리 주소를 바이트 단위로 정의하는 8비트 필드
      - 두 수준의 하드웨어 주소가 필요한 경우에는 reserved 필드가 두 번째 주소의 길이를 규정하기 위해 사용됨
      - ATM 네트워크의 경우, 20의 값을 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td colspan="2">Protocol Type</td></tr>
<tr><td>Sender Hardware Length</td><td>Reserved</td><td colspan="2">Operation</td></tr>
<tr><td>Sender Protocol Length</td><td>Target Hardware Length</td><td>Reserved</td><td>Target Protocol Length</td></tr>
<tr><td colspan="4">Sender hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Sender Protocol address</td></tr>
<tr><td colspan="4">Target hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Target protocol address</td></tr>
</table>

ARP

- ATMARP (6/7)
  - 패킷 구조
    - 송신자 하드웨어 주소(Sender Hardware Address) 필드
      - 송신자의 물리 주소를 정의하는 가변 길이 필드
      - ATM 네트워크의 경우에 20바이트의 길이를 가짐
    - 송신자 프로토콜 주소(Sender Protocol Address) 필드
      - 송신자의 프로토콜 주소를 정의하는 가변 길이 필드
      - IPv4의 경우에 4바이트의 길이를 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td colspan="2">Protocol Type</td></tr>
<tr><td>Sender Hardware Length</td><td>Reserved</td><td colspan="2">Operation</td></tr>
<tr><td>Sender Protocol Length</td><td>Target Hardware Length</td><td>Reserved</td><td>Target Protocol Length</td></tr>
<tr><td colspan="4">Sender hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Sender Protocol address</td></tr>
<tr><td colspan="4">Target hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Target protocol address</td></tr>
</table>

ARP

- ATMARP (7/7)
  - 패킷 구조
    - 타깃 하드웨어 주소(Target Hardware Address) 필드
      - 수신자의 물리 주소를 정의하는 가변 길이 필드
      - ATM 네트워크의 경우에 20바이트의 길이를 가짐
      - 요청 메시지는 비어 있고, 응답과 NACK 메시지에서 채워짐
    - 타깃 프로토콜 주소(Target Protocol Address) 필드
      - 수신자의 논리 주소를 정의하는 가변 길이 필드
      - IPv4의 경우에 4바이트의 길이를 가짐

<table>
<tr><td colspan="2">Hardware Type</td><td colspan="2">Protocol Type</td></tr>
<tr><td>Sender Hardware Length</td><td>Reserved</td><td colspan="2">Operation</td></tr>
<tr><td>Sender Protocol Length</td><td>Target Hardware Length</td><td>Reserved</td><td>Target Protocol Length</td></tr>
<tr><td colspan="4">Sender hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Sender Protocol address</td></tr>
<tr><td colspan="4">Target hardware address
(20 bytes)</td></tr>
<tr><td colspan="4">Target protocol address</td></tr>
</table>

ARP

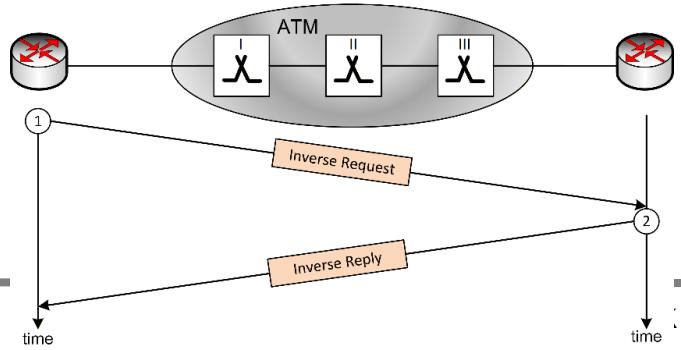
- ATMARP
  - 동작 (1/6)
    - PVC(Permanent Virtual Circuit) 연결 (1/2)
      - 정의
        - ATM 네트워크에서 두 종단점 사이에 가상 회선을 미리 설정하고, 이를 지속적으로 유지하여 사용하는 연결 방식
      - 특징
        - 네트워크 제공자가 각 스위치에 VPI와 VCI 값을 사전 등록
        - 두 라우터 간 주소 변환에 ATMARP 서버가 불필요
        - Inverse Request와 Inverse Reply 메시지를 이용하여 IP 주소와 ATM 물리 주소의 대응 관계를 설정함

*VPI(Virtual Path Identifier):
ATM 네트워크에서 가상 경로를 식별하기 위해 셀 헤더에 포함되는 식별자
*VCI (Virtual Channel Identifier)
ATM 네트워크에서 가상 채널을 식별하기 위해 셀 헤더에 포함되는 식별자

ARP

- ATMARP
  - 동작 (2/6)
    - PVC(Permanent Virtual Circuit) 연결 (2/2)
      - 주소 매핑 과정
        1. 라우터 1이 Inverse Request 메시지 전송
           - 송신자의 IP 주소와 ATM 물리 주소 포함
        2. 라우터 2가 Inverse Reply 메시지 전송
           - 응답 라우터의 IP 주소와 ATM 물리 주소 포함
        3. 두 라우터가 주소 매핑 정보 등록
           - 상대 라우터의 주소와 PVC 식별 정보의 대응 관계를 테이블에 저장
        4. IP 데이터그램 전송
           - 테이블에서 PVC 정보를 확인하고 데이터그램을 ATM 셀로 캡슐화하여 전송

Two routers connected through PVC

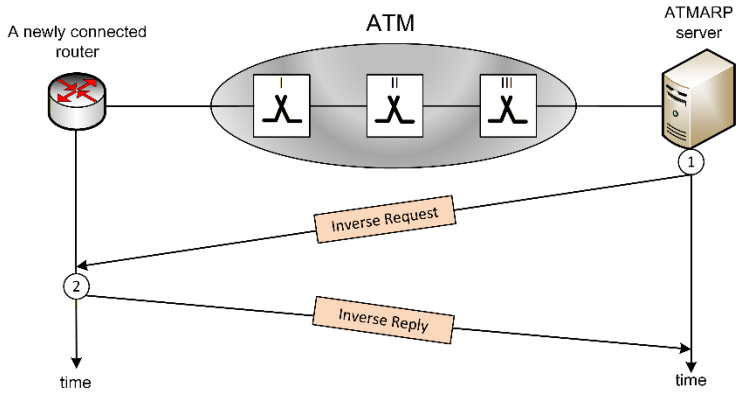


ARP

- ATMARP
  - 동작 (3/6)
    - SVC(Switched Virtual Circuit) 연결 (1/4)
      - 정의
        - ATM 네트워크에서 통신이 필요할 때마다 두 종단점 사이에 새로운 가상 회선을 설정하는 연결 방식
      - 특징
        - 송신 측 라우터가 목적지 측 라우터의 ATM 물리 주소를 알아야 연결 설정 가능
        - 각 라우터는 ATMARP 클라이언트로 동작하고, 한 컴퓨터는 ATMARP 서버로 동작
        - ATMARP 서버를 통해 IP 주소에 대응하는 ATM 물리 주소 확인

ARP

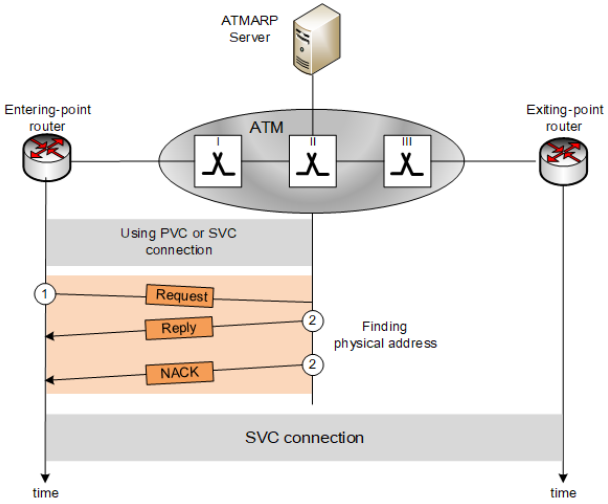
- ATMARP
  - 동작 (4/6)
    - SVC(Switched Virtual Circuit) 연결 (2/4)
      - 주소 정보 등록 과정
        1. 새로운 라우터가 ATMARP 서버와 연결
           - ATM 네트워크에 접속한 라우터가 ATMARP 서버와 가상 회선을 설정함
        2. ATMARP 서버가 Inverse Request 메시지 전송
           - 새로 연결된 라우터의 IP 주소와 ATM 물리 주소를 요청함
        3. 라우터가 Inverse Reply 메시지 전송
           - 자신의 IP 주소와 ATM 물리 주소를 메시지에 담아 서버에 응답함
        4. ATMARP 서버가 주소 매핑 정보 등록
           - 수신한 IP 주소와 ATM 물리 주소의 대응 관계를 ATMARP 테이블에 저장함



ARP

- ATMARP
  - 동작 (5/6)
    - SVC(Switched Virtual Circuit) 연결 (3/4)
      - 주소 확인 과정

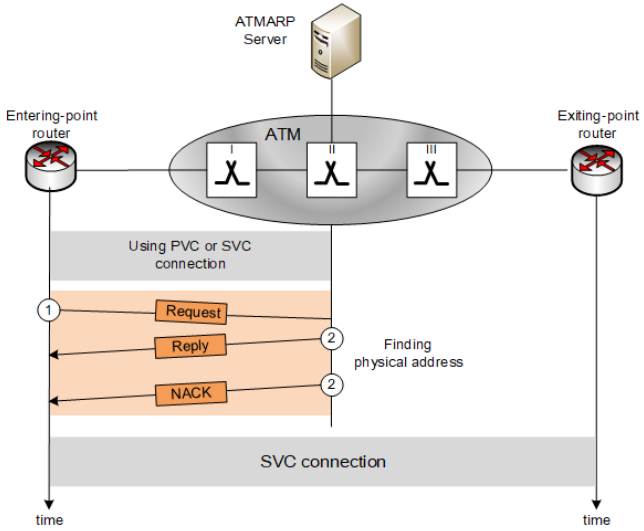
*NACK:
요청한 정보를 찾지 못했거나 요청을 정상적으로 처리할 수 없음을 알리는 부정 응답 메시지.



1. ATMARP 서버와 연결
   - 일반적으로 각 라우터와 ATMARP 서버 사이에는 PVC가 미리 설정됨
   - PVC가 없으면 서버의 물리 주소를 이용하여 ATMARP 메시지 교환용 SVC를 임시로 설정함
2. 목적지 측 라우터의 물리 주소 확인
   - 송신 측 라우터가 목적지 측 라우터의 IP 주소를 담은 ATMARP Request를 서버에 전송함
   - 서버가 ATMARP 테이블에서 해당 IP 주소를 검색함
     - 주소를 찾으면 ATMARP Reply로 물리 주소를 응답하고, 찾지 못하면 ATMARP NACK를 전송함
     - NACK를 받은 경우 해당 IP 데이터그램은 폐기됨

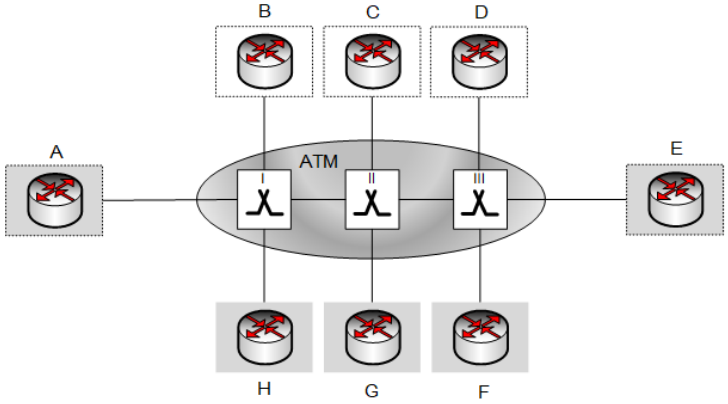
ARP

- ATMARP
  - 동작 (6/6)
    - SVC(Switched Virtual Circuit) 연결 (4/4)
      - 주소 확인 과정
        3. SVC 설정
           - 송신 측 라우터가 확인한 물리 주소를 이용하여 목적지 측 라우터와 SVC 설정을 요청함
           - ATM 네트워크가 두 라우터의 물리 주소를 이용하여 가상 회선을 생성함
           - 각 ATM 스위치가 셀 전달에 필요한 경로 정보를 자신의 테이블에 등록함
           - 설정된 SVC는 송신 측 라우터가 연결 해제를 요청할 때까지 유지됨



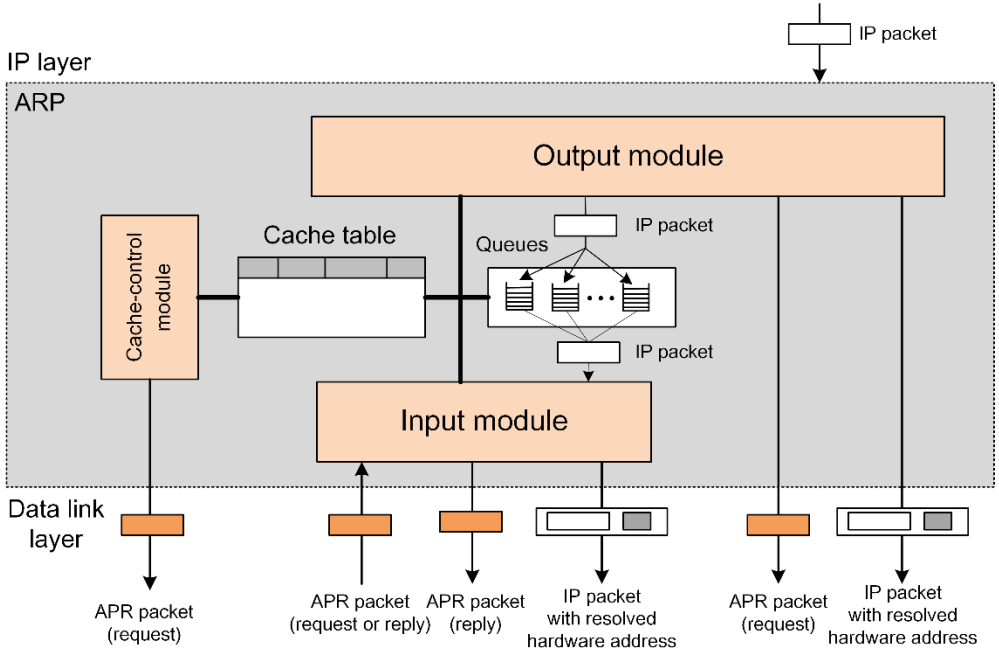
ARP

- ATMARP
  - 논리적 IP 서브넷(LIS, Logical IP Subnet)
    - 정의
      - 하나의 ATM 네트워크를 물리적으로 분리하지 않고 여러 개의 IP 서브넷으로 논리적으로 구분한 네트워크 단위
    - 특징
      - 동일한 LIS의 라우터는 같은 네트워크 접두사와 서브넷 마스크 사용
      - 하나의 라우터가 둘 이상의 LIS에 동시에 소속 가능
      - 동일한 LIS 간에는 중간 IP 라우터 없이 직접 통신
      - 서로 다른 LIS 간에는 두 LIS에 모두 연결된 라우터를 경유
      - 각 LIS마다 별도의 ATMARP 서버 필요



ARP

- ARP 패키지
  - 정의
    - IP 주소에 대응하는 물리 주소를 확인하고, IP 패킷을 데이터 링크 계층으로 전달하기 위한 ARP 처리 소프트웨어 모듈 집합
  - 구성 요소
    - 캐시 테이블(Cache Table)
    - 큐(Queues)
    - 출력 모듈(Output Module)
    - 입력 모듈(Input Module)
    - 캐시 제어 모듈 (Cache-Control Module)



ARP

- ARP 패키지 – 구성요소 (1/6)
  - 캐시 테이블(Cache Table)
    - 정의
      - ARP를 통해 확인한 IP 주소와 물리 주소의 대응 관계를 일정 시간 저장하는 테이블
    - 특징
      - 동일한 목적지로 패킷을 보낼 때 저장된 주소를 재사용하여 반복적인 ARP 수행 방지
      - 여러 개의 항목으로 구성된 배열 형태로 구현
      - 저장 공간이 제한되어 있어 주소 정보를 영구적으로 보관하지 않음
      - 각 항목의 처리 상태를 FREE, PENDING, RESOLVED로 구분

ARP

- ARP 패키지 – 구성요소 (2/6)
  - 캐시 테이블(Cache Table)

Field	Description
State	엔트리의 상태를 보여줌 • FREE: 엔트리의 수명이 만료되었음을 의미하고, 다른 엔트리를 위하여 사용될 수 있음 • PENDING: ARP 요청을 보냈으나 응답을 기다리는 상태 • RESOLVED: IP 주소에 대응하는 물리 주소 확인이 완료된 상태
Hardware Type	ARP가 수행되고 있는 네트워크의 유형을 정의함 (ARP 패킷에서의 하드웨어 유형과 같음)
Protocol Type	프로토콜을 정의함
Hardware Length	물리 주소의 길이를 바이트 단위로 정의함
Protocol Length	논리 주소의 길이를 바이트 단위로 정의함
Interface Number	다수의 인터페이스를 가진 호스트에서, 각 네트워크 인터페이스를 식별할 번호를 보여줌
Queue Number	같은 목적지 IP 주소로 전송될 패킷들을 모아놓은 큐를 식별하기 위한 번호를 보여줌
Attempts	해당 엔트리에 대해 ARP 요청을 전송한 횟수를 보여줌
Time-Out	해당 엔트리의 TTL을 초단위로 환산한 값을 보여줌
Hardware Address	목적지 하드웨어의 주소를 보여줌 ARP 응답을 통해 주소 변환이 이루어지기 전까지 비어 있음
Protocol Address	목적지 IP 주소를 보여줌

ARP

- ARP 패키지 – 구성요소 (3/6)
  - 큐(Queues)
    - ARP가 하드웨어 주소를 얻기 전 IP 데이터그램이 저장되는 자료구조
    - 한 목적지당 하나씩 유지됨
    - 출력 모듈은 해결되지 않은 패킷을 해당 큐에 보냄
    - 입력 모듈은 큐에서 패킷을 해결된 물리 주소와 같이 가져와 데이터 링크 계층으로 전달함

• ARP 패키지 – 구성요소 (4/6)

- 출력 모듈
  - 정의
    - IP 계층에서 받은 패킷의 목적지 MAC 주소를 ARP 캐시에서 확인하고, 주소 정보가 없으면 ARP 요청을 수행하는 모듈
  - 동작 과정

```
ARP_Output_Module ()
{
    Sleep until an IP packet is received from IP software.
    Check cache table for an entry corresponding to the destination of IP packet.
    if (entry is found)  {
            if (the state is RESOLVED) {
                    Extract the value of the hardware address from the entry.
                    Send the packet and the hardware address to data link layer.
                    Return
     }   //end if
     if (the state is PENDING) {
                    Enqueue the packet to the corresponding queue.
                    Return
            }       //end if
     }              //end if
     if(entry is not found) {
            Create a cache entry with state set to PENDING and ATTEMPTS set to 1.
            Create a queue.
            Enqueue the packet.
            Send and ARP request.
            Return
     }              //end if
}                   //end module
```

1. IP 계층으로부터 IP 패킷이 수신될 때까지 대기함
2. 패킷이 수신되면 목적지 IP 주소에 해당하는 캐시 항목을 확인함
3. 캐시 상태가 RESOLVED이면 물리 주소를 가져와 패킷과 함께 데이터 링크 계층으로 전달함
4. 캐시 상태가 PENDING이면 패킷을 해당 목적지의 큐에 저장함
5. 캐시 항목이 없으면 PENDING 상태의 항목과 큐를 생성하고 패킷을 저장한 후 ARP 요청을 전송함

ARP

- ARP 패키지 – 구성요소 (5/6)
  - 입력 모듈
    - 정의
      - 데이터 링크 계층에서 수신한 ARP 요청·응답 패킷을 처리하고, 캐시 테이블을 갱신하거나 필요한 ARP 응답을 전송하는 모듈
    - 동작 과정

```
ARP_Input_Module ()
{
    Sleep until an ARP packet (request or reply) arrives.
    Check the cache table to find the corresponding entry.
    if (found)  {
            Update the entry.
            if(the state is PENDING) {
                    While (the queue is not empty) {
                        Dequeue one packet.
                        Send the packet and the hardware address.
                    } //end if
             }   //end if
        } //end if
        if (not found) {
                    Create and entry.
                    Add the entry to the table.
        }
        if (the packet is a request)
        {
                    Send an ARP reply.
         }          //end if
        Return
}  //end module
```

1. ARP 요청 또는 응답 패킷이 수신될 때까지 대기함
2. 패킷이 수신되면 캐시 테이블에서 해당 항목이 존재하는지 확인함
3. 캐시 항목이 존재하면 항목을 갱신하고, 상태가 PENDING이면 큐의 패킷을 하나씩 꺼내 물리 주소와 함께 전송함
4. 캐시 항목이 존재하지 않으면 새로운 항목을 생성하여 캐시 테이블에 추가함
5. 수신한 패킷이 ARP 요청이면 ARP 응답을 전송함

ARP

- ARP 패키지 – 구성요소 (6/6)
  - 캐쉬 제어 모듈
    - 정의
      - 캐시 테이블의 항목 상태와 유효 시간을 관리하는 모듈
    - 동작 과정

1. 주기 타이머가 만료될 때까지 대기한 후 캐시 테이블의 모든 항목을 순서대로 확인함
2. 상태가 FREE이면 별도의 처리 없이 다음 항목으로 넘어감
3. 상태가 PENDING이면 시도 횟수(Attempts)를 1 증가시키고, 최대 횟수를 초과하면 상태를 FREE로 변경한 후 해당 큐를 삭제함
4. PENDING 상태에서 최대 시도 횟수를 초과하지 않았으면 ARP 요청을 다시 전송함
5. 상태가 RESOLVED이면 타임아웃(Time-out)을 감소시키고, 값이 0 이하가 되면 상태를 FREE로 변경하고 해당 큐를 삭제함

```
ARP_Cache_Control_Module ( )
{
    Sleep until the periodic timer matures.
    Repeat for every entry in the cache table
    {
            if (the state is FREE) {
                    Continue.
    } // end if
    if (the state is PENDING) {
            Increment the value of attempts by 1.
            if (attempts greater than maximum) {
                    Change the state to FREE.
                    Destroy the corresponding queue.
            } //end if
            else {
                    Send and ARP request
            } //end else
            continue.
        } //end if
        if (the state is RESOLVED) {
                Decrement the value of time-out.
                if (time-out less than or equal 0) {
                  Change the state to FREE.
                  Destroy the corresponding queue.
                } // end if
            } //end if
    } //end repeat
    Return.
} // end module
```

ARP

- ARP 패키지 (1/3)
  - 예제 8.2

ARP 출력 모듈은 IP 계층으로부터 목적지 주소가 114.5.7.89인 IP 데이터그램을 수신한다.해당 표는 캐쉬 테이블의 초기 값이다.캐쉬 테이블의 변화를 관찰하라.

- 캐시 테이블을 검사한 결과, 해당 주소를 위한 항목이 RESOLVED 상태로 존재함
- 하드웨어 주소 457342ACAE32를 꺼내어 패킷과 이 주소를 데이터 링크 계층에 전달함

State	Queue	Attempts	Time-Out	Protocol Addr.	Hardware Addr.
R	5	-	900	180.3.6.1	ACAE32457342
P	2	2	-	129.34.4.8	-
P	14	5	-	201.11.56.7	-
R	8	-	450	114.5.7.89	457342ACAE32
P	12	1	-	220.55.5.7	-
F	-	-	-	-	-
R	9	-	60	19.1.7.82	4573E3242ACA
P	18	3	-	188.11.8.71	-

ARP

- ARP 패키지 (2/3)
  - 예제 8.3

20초 후 ARP 출력 모듈이 IP 계층으로부터 목적지 주소가 116.1.7.22인 IP 데이터그램을 수신한다. 해당 상황에 대하여 동작을 분석하라.

- 캐쉬 테이블을 검사한 결과, 목적지 주소가 동일한 항목이 존재하지 않음
  - 모듈은 새 항목을 테이블에 추가하고, 해당 항목의 상태를 PENDING으로 설정한 후 시도 값을 1로 설정함
  - 모듈은 큐를 새로 만들어 패킷을 해당 큐에 넣고 ARP 요청을 데이터 링크 계층을 통하여 목적지로 전송함

State	Queue	Attempts	Time-Out	Protocol Addr.	Hardware Addr.
R	5	-	900	180.3.6.1	ACAE32457342
P	2	2	-	129.34.4.8	-
P	14	5	-	201.11.56.7	-
R	**8**	**-**	**450**	**114.5.7.89**	**457342ACAE32**
P	12	1	-	220.55.5.7	-
F	-	-	-	-	-
R	9	-	60	19.1.7.82	4573E3242ACA
P	18	3	-	188.11.8.71	-

→

State	Queue	Attempts	Time-Out	Protocol Addr.	Hardware Addr.
R	5	-	900	180.3.6.1	ACAE32457342
P	2	2	-	129.34.4.8	-
P	14	5	-	201.11.56.7	-
R	8	-	450	114.5.7.89	457342ACAE32
P	12	1	-	220.55.5.7	-
P	**23**	**1**	**-**	**116.1.7.22**	**-**
R	9	-	60	19.1.7.82	4573E3242ACA
P	18	3	-	188.11.8.71	-

ARP

- ARP 패키지 (3/3)
  - 예제 8.4

15초 후 ARP 입력 모듈은 타겟 프로토콜(IP)주소 188.11.8.71을 가진 ARP 패킷을 수신한다. 해당 상황에 대하여 동작을 분석하라.

- 캐쉬 테이블을 검사한 결과, 해당 주소를 가지고 있음을 확인할 수 있음
  - 모듈은 타깃 하드웨어 주소 E34573242ACA를 삽입함
  - 모듈은 큐 18에 접근하여 해당 큐 속에서 대기 중인 모든 패킷을 데이터 링크 계층으로 하나씩 전송함

State	Queue	Ateempt	Time-Out	Protocol Addr.	Hardware Addr.
R	5	-	900	180.3.6.1	ACAE32457342
P	2	2	-	129.34.4.8	-
P	14	5	-	201.11.56.7	-
R	8	-	450	114.5.7.89	457342ACAE32
P	12	1	-	220.55.5.7	-
P	**23**	**1**	**-**	**116.1.7.22**	**-**
R	9	-	60	19.1.7.82	4573E3242ACA
P	18	3	-	188.11.8.71	-

→

State	Queue	Ateempt	Time-Out	Protocol Addr.	Hardware Addr.
R	5	-	900	180.3.6.1	ACAE32457342
P	2	2	-	129.34.4.8	-
P	14	5	-	201.11.56.7	-
R	8	-	450	114.5.7.89	457342ACAE32
P	12	1	-	220.55.5.7	-
F	23	1	-	116.1.7.22	-
R	9	-	60	19.1.7.82	4573E3242ACA
P	**18**	**3**	**-**	**188.11.8.71**	**E34573242ACA**

목 차

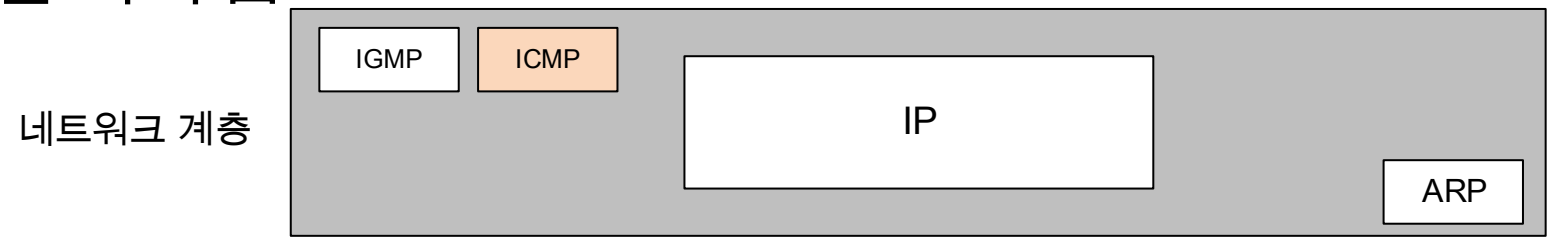
- ARP(Address Resolution Protocol)
- ICMPv4(Internet Control Message Protocol Version 4)
- 모바일 IP(Mobile IP)

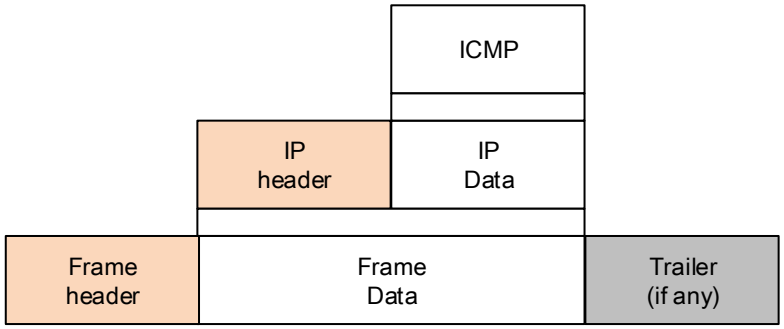
ICMPv4

- 개요
  - IPv4는 최선 노력 전달(Best-effort Delivery) 방식으로 동작하여 패킷 전달 과정의 오류 원인이나 네트워크 상태에 대한 피드백 기능이 부족함
  - ICMP(Internet Control Message Protocol)는 이러한 문제점을 보완하기 위해 등장함
- 정의
  - IPv4의 패킷 전달 과정에서 오류 및 네트워크 상태 정보를 전달하는 네트워크 계층 프로토콜

ICMPv4

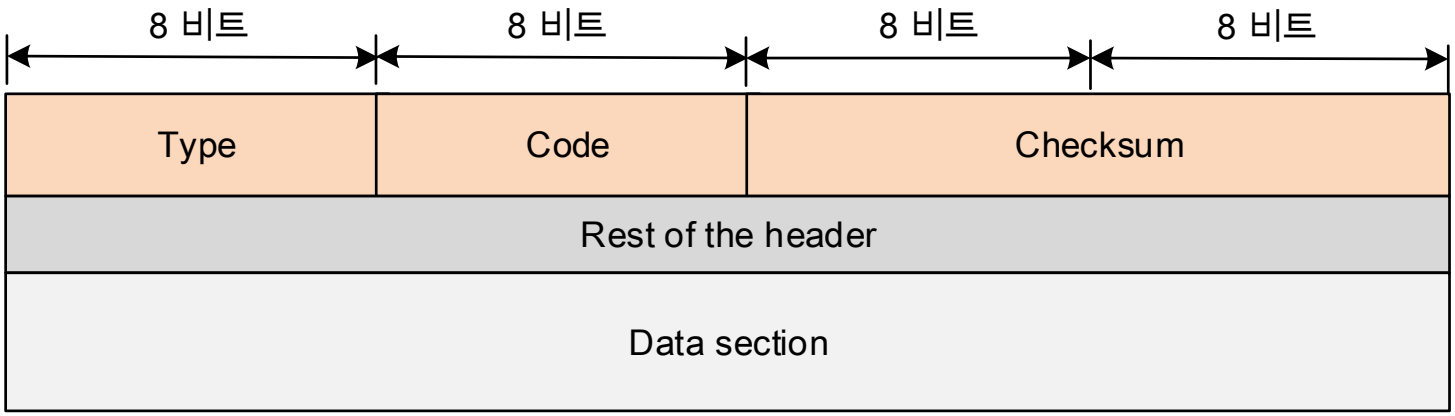
- 특징
  - IPv4와 함께 동작하는 네트워크 계층 프로토콜임
  - ICMP 메시지는 IPv4 패킷의 데이터 영역에 캡슐화 (Encapsulation) 되어 전송됨
    - 해당 IP 패킷 헤더의 프로토콜 필드 값을 1로 하여 ICMP 메시지임을 나타냄





ICMPv4

- 메시지 형식(Message Format)
  - 메시지는 8바이트의 헤더와 가변 길이의 데이터 부분을 가짐
  - 헤더의 일반 형식은 메시지 별로 다르지만 처음 4바이트는 전부 공통임
  - 나머지 헤더(Rest of the header)와 데이터 영역(Data Section)은 메시지 유형에 따라 다르게 구성됨



ICMPv4

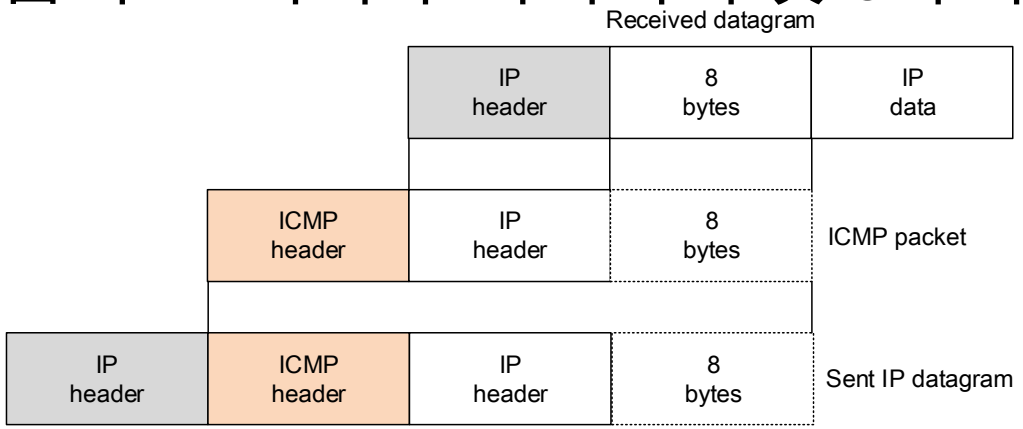
- 메시지 유형(Message Type)
  - 오류 보고 메시지는 라우터나 호스트가 IP 패킷을 처리하는 도중 탐지하는 문제를 보고함
  - 질의 메시지는 호스트나 네트워크 관리자가 라우터나 다른 호스트로부터 특정 정보를 획득하기 위해 사용함

분류	유형	메시지
오류 보고 메시지 (Error-reporting messages)	3	목적지 도달 불가
	4	발신지 억제
	11	시간 경과
	12	매개변수 문제
	5	재지정
질의 메시지 (Query messages)	8 / 0	에코 요청 / 응답
	13 / 14	타임스탬프 요청 / 응답
	10 / 9	라우터 간청 / 광고
	15 / 16	정보 요청 / 응답
	17 / 18	주소 마스크 요청 / 응답

ICMPv4

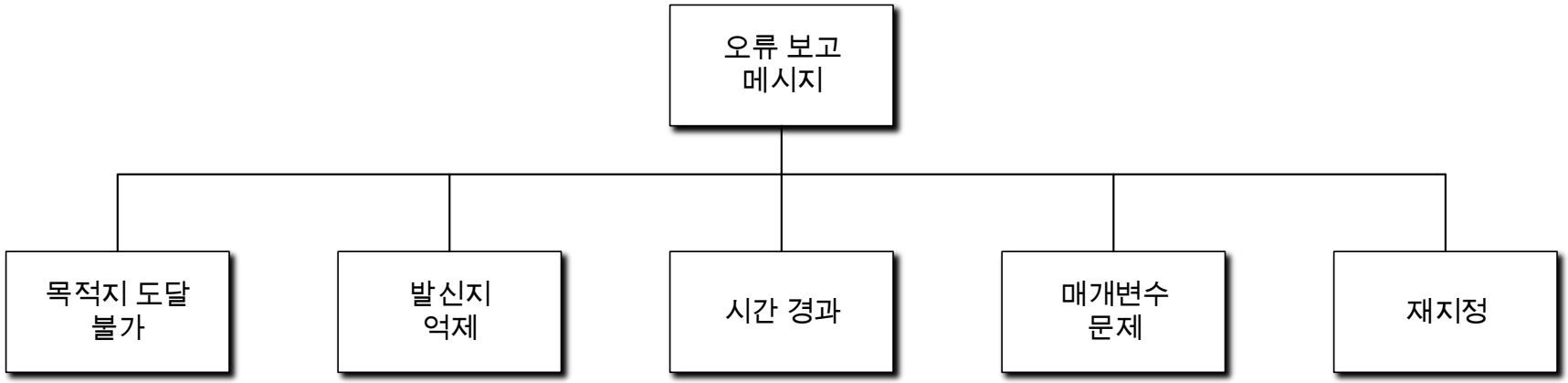
- 오류 보고 메시지 (1/10)
  - 정의

    라우터나 목적지 호스트가 IP 패킷을 처리하는 과정에서 발생한 문제를 최초 발신지에 알리는 ICMP 메시지
  - 특징
    - 발생한 오류를 수정하지 않고 보고만 함
    - 오류 메시지는 항상 원본 데이터그램의 최초 발신지로 전송됨
    - 오류 보고 메시지의 데이터 영역에는 오류가 발생한 원본 데이터그램의 IP 헤더와 데이터의 첫 8바이트가 포함됨



ICMPv4

- 오류 보고 메시지 (2/10)
  - 종류
    - 목적지 도달 불가(Destination Unreachable)
    - 발신지 억제(Source Quench)
    - 시간 경과(Time Exceeded)
    - 매개변수 문제(Parameter Problem)
    - 재지정(Redirect)



ICMPv4

- 오류 보고 메시지 (3/10)
  - 목적지 도달 불가(Destination Unreachable) 메시지
    - 정의
      - 라우터가 데이터그램을 라우팅할 수 없거나 목적지 호스트가 데이터그램을 전달할 수 없을 때, 이를 발신지에 알리기 위해 생성되는 메시지
    - 특징
      - 목적지에 도달하지 못한 데이터그램은 폐기됨
      - Type 값은 3이며, Code 0~15를 통해 폐기된 구체적인 이유를 나타냄
        - e.g., Code 0: 목적지 네트워크에 도달할 수 없음
      - 목적지 도달 불가 메시지를 받지 않았다고 해서 데이터그램이 정상적으로 전달되었다는 의미는 아님

Type: 3	Code: 0 to 15	Checksum
Unused (All 0s)		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

ICMPv4

- 오류 보고 메시지 (4/10)
  - 목적지 도달 불가(Destination Unreachable) 메시지
    - 코드 (1/2)
      - 코드 2, 3은 목적지 호스트에 의해서 생성되고 나머지 코드는 라우터에 의해 생성됨

코드 값	의미	설명
0	네트워크 도달 불가	목적지 네트워크는 존재하지만 현재 해당 네트워크로 전달할 수 없음
1	호스트에 도달 불가	목적지 호스트는 존재하지만 현재 해당 호스트로 전달할 수 없음
2	프로토콜 도달 불가	목적지 호스트에서 지정된 상위 계층 프로토콜을 사용할 수 없음
3	포트 도달 불가	목적지 포트에서 해당 응용 프로그램이 실행되고 있지 않음
4	DF 설정으로 인한 단편화 불가	전달을 위해 단편화가 필요하지만 DF가 설정되어 있어 단편화 할 수 없음
5	발신지 라우팅 수행 불가	발신지가 지정한 경로의 하나 이상의 라우터를 거칠 수 없음
6	알려지지 않은 목적지 네트워크	라우터가 목적지 네트워크에 대한 정보를 가지고 있지 않음
7	알려지지 않은 목적지 호스트	라우터가 목적지 호스트의 존재에 대한 정보를 가지고 있지 않음

ICMPv4

- 오류 보고 메시지 (5/10)
  - 목적지 도달 불가(Destination Unreachable) 메시지
    - 코드 (2/2)
      - 코드 2, 3은 목적지 호스트에 의해서 생성되고 나머지 코드는 라우터에 의해 생성됨

코드 값	의미	설명
8	발신지 호스트 고립	발신지 호스트가 네트워크에서 격리되어 통신할 수 없음
9	네트워크 차단	관리 정책에 의해 목적지 네트워크와의 통신이 금지됨
10	호스트 차단	관리 정책에 의해 목적지 호스트와의 통신이 금지됨
11	서비스 유형에 대해 네트워크 도달 불가	요청한 서비스 유형으로는 목적지 네트워크까지 전달할 수 없음
12	서비스 유형에 대해 호스트 도달 불가	요청한 서비스 유형으로는 목적지 호스트까지 전달할 수 없음
13	관리상의 이유로 차단	관리자가 설정한 필터링 정책으로 인해 패킷 전달이 차단됨
14	호스트 우선순위 위반	요청한 우선순위가 목적지 호스트에서 허용되지 않음
15	우선순위 차단	패킷의 우선순위가 네트워크에서 요구하는 최소 우선순위보다 낮음

ICMPv4

- 오류 보고 메시지 (6/10)
  - 발신지 억제(Source Quench) 메시지
    - 정의
      - 송신자에게 네트워크 혼잡으로 인해 데이터그램이 폐기되었음을 알리고, 송신 속도를 줄이도록 요청하는 메시지
    - 특징
      - 혼잡이 해소되었음을 송신자에게 알려주는 별도의 메커니즘은 없음
      - 송신자는 발신지 억제 메시지가 더 이상 수신되지 않을 때까지 전송 속도를 낮춤
      - 폐기된 데이터그램 하나당 하나의 발신지 억제 메시지를 전송함
      - 별도의 세부 원인 구분이 없어 Code 값은 0만 사용함

Type: 4	Code: 0	Checksum
Unused (All 0s)		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

ICMPv4

- 오류 보고 메시지 (7/10)
  - 시간 경과(Time Exceeded) 메시지
    - 정의
      - 데이터그램의 TTL(Time To Live)이 0이 되거나, 정해진 시간 내에 모든 단편이 도착하지 않아 데이터그램이 폐기되었음을 송신자에게 알리는 메시지
    - 특징
      - Code 값으로 시간 경과가 발생한 원인을 구분함
        - e.g., Code 0: TTL 값이 0이 되어 데이터그램이 폐기됨
      - 오류가 발생한 데이터그램 또는 단편은 폐기
      - 시간 경과 메시지는 최초 송신자에게 전송됨

Type: 11	Code: 0 or 1	Checksum
Unused (All 0s)		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

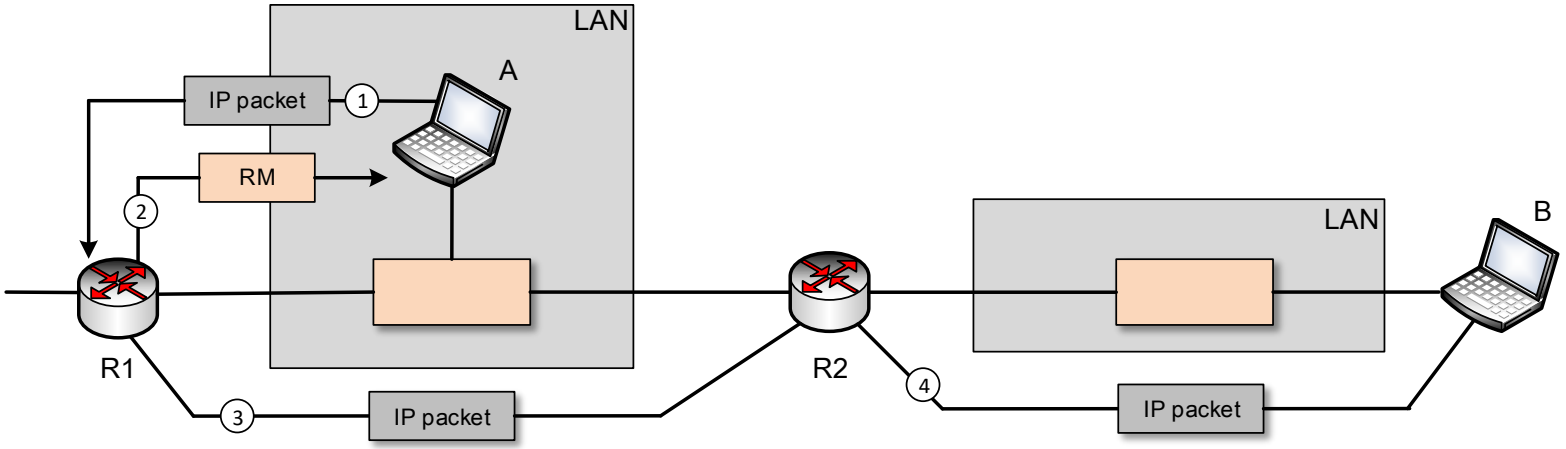
ICMPv4

- 오류 보고 메시지 (8/10)
  - 매개변수 문제(Parameter Problem) 메시지
    - 정의
      - 데이터그램 헤더의 필드에 잘못되거나 모호한 값이 있거나, 필요한 옵션이 누락된 경우 전송되는 메시지
    - 특징
      - Code 값으로 데이터그램이 폐기된 원인을 구분함
        - e.g., Code 0: 헤더 필드에 오류 또는 모호한 값이 존재함
      - Code 0인 경우 Pointer 필드가 문제가 발생한 바이트의 위치를 나타냄
      - 라우터 또는 목적지 호스트가 생성할 수 있음

Type: 12	Code: 0 or 1	Checksum
Pointer	Unused (All 0s)	
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

ICMPv4

- 오류 보고 메시지 (9/10)
  - 재지정(Redirect) 메시지 (1/2)
    - 정의
      - 호스트가 데이터그램을 적절하지 않은 라우터로 전송했을 때, 더 적절한 라우터를 사용하도록 송신자에게 알리는 메시지
    - 특징
      - 호스트의 라우팅 테이블을 갱신하는 데 사용됨
      - Code 값으로 재지정 대상의 종류를 구분함
      - 메시지에 더 적절한 라우터의 IP 주소가 포함됨



ICMPv4

- 오류 보고 메시지 (10/10)
  - 재지정(Redirect) 메시지 (2/2)
    - 코드

Type: 5	Code: 0 to 3	Checksum
IP address of the target router		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

코드 값	의미	설명
0	네트워크 경로 재지정	목적지 네트워크로 가기 위한 더 적절한 라우터가 존재함
1	호스트 경로 재지정	목적지 호스트로 가기 위한 더 적절한 라우터가 존재함
2	특정 서비스 유형에 대한 네트워크 경로 재지정	지정된 서비스 유형으로 목적지 네트워크에 가기 위한 더 적절한 라우터가 존재함
3	특정 서비스 유형에 대한 호스트 경로 재지정	지정된 서비스 유형으로 목적지 호스트에 가기 위한 더 적절한 라우터가 존재함

ICMPv4

- 질의 메시지 (1/5)
  - 정의
    - 호스트나 네트워크 관리자가 다른 호스트 또는 라우터로부터 특정 정보를 얻거나 네트워크 상태를 진단하기 위해 사용하는 ICMP 메시지
  - 특징
    - 대부분 요청(Request)과 응답(Reply)이 한 쌍으로 동작함
    - 요청 메시지를 받은 목적지 노드는 정해진 형식의 응답 메시지를 반환함
    - 네트워크의 상태 확인 및 진단에 사용됨

ICMPv4

- 질의 메시지 (2/5)
  - 에코 요청과 응답(Echo Request and Reply) 메시지
    - 정의
      - 두 호스트 또는 라우터 간에 IP 수준의 통신이 가능한지 확인하기 위해 주고받는 질의 메시지
    - 특징
      - PING에서 호스트의 도달 가능성(Reachability)을 검사할 수 있음
      - 에코 요청을 받은 호스트 또는 라우터가 에코 응답을 생성하여 최초 송신자에게 반환함
      - 식별자(Identifier)와 순서 번호(Sequence number)를 이용해 요청과 응답을 구분할 수 있음

Type: 8 or 0	Code: 0	Checksum
Identifier		Sequence number
Optional data Sent by the request message; repeated by the reply message		

ICMPv4

- 질의 메시지 (3/5)
  - 타임스탬프 요청과 응답(Timestamp Request and Reply) 메시지 (1/3)
    - 정의
      - 두 호스트 또는 라우터 간 IP 데이터그램의 왕복 시간(RTT)을 측정하기 위해 주고받는 질의 메시지
    - 특징
      - 요청은 Type 13, 응답은 Type 14, Code는 0을 사용함
      - 타임스탬프 값을 이용해 편도 시간과 왕복 시간을 계산할 수 있으며, 조건에 따라 두 장치의 시계 동기화에도 사용할 수 있음
      - Original / Receive / Transmit Timestamp의 3개 타임스탬프 필드를 사용함
      - 각 타임스탬프는 32비트이며, UTC 자정부터 경과한 시간을 밀리초(ms) 단위로 나타냄

Type: 13 or 14	Code: 0	Checksum
Identifier		Sequence number
Original timestamp		
Receive timestamp		
Transmit timestamp		

ICMPv4

- 질의 메시지 (4/5)
  - 타임스탬프 요청과 응답(Timestamp Request and Reply) 메시지 (2/3)
    - 계산 방법
      - 송신 시간 = 수신 타임스탬프 – 오리지널 타임스탬프
      - 수신 시간 = 패킷이 돌아온 시간 – 전송 타임스탬프
      - 왕복 시간 = 송신 시간 + 수신 시간
      - 예제 9.1

타임 스탬프의 요청과 응답 메시지를 통해 다음과 같은 정보를 얻었을 때, 왕복 시간을 구하시오.

- 원래 타임스탬프의 값: 46 전달 타임스탬프의 값:59
  수신 타임 스탬프의 값: 60 패킷 도착 시각: 67

- 송신 시간 = 수신 타임스탬프 – 오리지널 타임스탬프 $= 59 - 46 = 13$ms
- 수신 시간 = 패킷이 돌아온 시간 – 전송 타임스탬프 $= 67 - 60 = 7$ms
- 왕복 시간 = 송신 시간 + 수신 시간 $= 13 + 7 = 20$ms

ICMPv4

- 질의 메시지 (5/5)
  - 타임스탬프 요청과 응답(Timestamp Request and Reply) 메시지 (3/3)
    - 계산 방법
      - 시간 차 = 수신 타임스탬프 – (원래 타임스탬프 필드 + 편도 시간 구간)
      - 편도 시간=왕복 시간/2
      - 예제 9.2

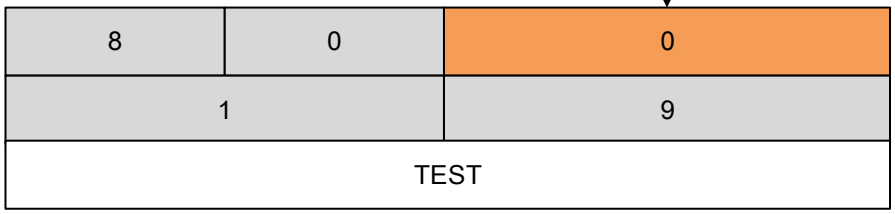
타임 스탬프의 요청과 응답 메시지를 통해 다음과 같은 정보를 얻었을 때, 왕복 시간을 구하시오.

- 원래 타임스탬프의 값: 46 전달 타임스탬프의 값: 59
  수신 타임 스탬프의 값: 60 패킷 도착 시각: 67

- 편도 시간 = 왕복 시간/2 = 20/2 = 10ms
- 시간 차 = 수신 타임스탬프 – (원래 타임스탬프 필드 + 편도 시간 구간)
  = 60 – (46 + 10) = 4ms

ICMPv4

- 체크섬 (1/2)
  - 정의
    - ICMP 메시지의 전송 중 오류 여부를 확인하기 위해 사용하는 오류 검출 필드
  - 계산 과정
    1. 체크섬 필드를 0으로 만듦
    2. 헤더와 데이터에 대해 16비트 단위의 합을 구함
    3. 합에 1의 보수를 취하여 체크섬을 구함
    4. 구한 체크섬을 체크섬 필드에 저장함



8 & 0	→	00001000	00000000
0	→	00000000	00000000
1	→	00000000	00000001
9	→	00000000	00001001
T & E	→	01010100	01000101
S & T	→	01010011	01010100
Sum	→	10101111	10100011
Checksum	→	01010000	01011100

ICMPv4

- 체크섬 (2/2)
  - 검사 과정
    1. 헤더와 데이터에 대해 16비트 단위의 합을 구함
    2. 합의 보수를 구함
    3. 2단계의 결과가 16개의 0이면 메시지는 받아들여지고 그렇지 않으면 메시지를 폐기함

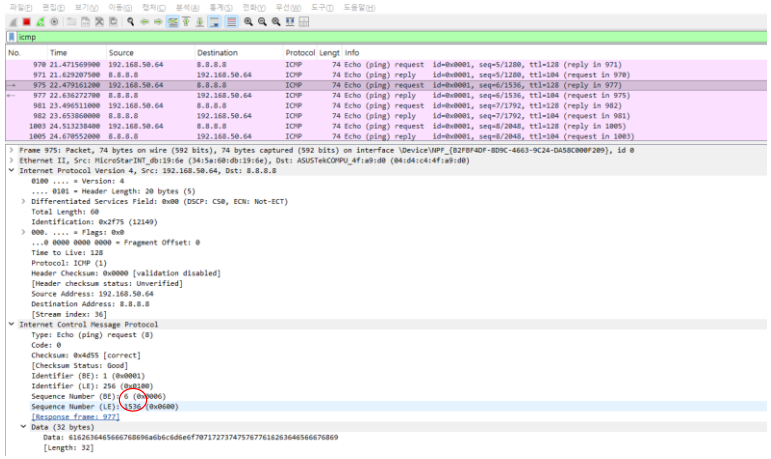
8	0	20572
1		9
TEST		

8 & 0	→	00001000	00000000
20572	→	01010000	01011100
1	→	00000000	00000001
9	→	00000000	00001001
T & E	→	01010100	01000101
S & T	→	01010011	01010100
Sum	→	11111111	11111111
Result	→	00000000	00000000

ICMPv4

- 디버깅 도구 (1/4)
  - PING (1/2)
    - 정의
      - 호스트가 정상적으로 동작하고 응답 가능한지 확인하기 위해 사용하는 네트워크 진단 프로그램
    - 특징
      - 송신 호스트가 Echo Request 메시지를 전송함
      - 목적지가 정상 동작 중이면 Echo Reply 메시지로 응답함
      - Sequence Number는 0부터 시작하여 메시지를 보낼 때마다 증가함

```
C:\Users\ms612>ping 8.8.8.8

Ping 8.8.8.8 32바이트 데이터 사용:
8.8.8.8의 응답: 바이트=32 시간=157ms TTL=104
8.8.8.8의 응답: 바이트=32 시간=157ms TTL=104
8.8.8.8의 응답: 바이트=32 시간=157ms TTL=104
8.8.8.8의 응답: 바이트=32 시간=157ms TTL=104

8.8.8.8에 대한 Ping 통계:
    패킷: 보냄 = 4, 받음 = 4, 손실 = 0 (0% 손실),
왕복 시간(밀리초):
    최소 = 157ms, 최대 = 157ms, 평균 = 157ms
```



ICMPv4

- 디버깅 도구 (2/4)
  - PING (2 /2)

icmp

No.	Time	Source	Destination	Protocol	Lengt	Info
970	21.471569900	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=5/1280, ttl=128 (reply in 971)
971	21.629207500	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=5/1280, ttl=104 (request in 970)
975	22.479161200	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=6/1536, ttl=128 (reply in 977)
977	22.636272700	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=6/1536, ttl=104 (request in 975)
981	23.496511000	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=7/1792, ttl=128 (reply in 982)
982	23.653860000	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=7/1792, ttl=104 (request in 981)
1003	24.513238400	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=8/2048, ttl=128 (reply in 1005)
1005	24.670552000	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=8/2048, ttl=104 (request in 1003)

```
    Protocol: ICMP (1)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.50.64
    Destination Address: 8.8.8.8
    [Stream index: 36]
Internet Control Message Protocol
    Type: Echo (ping) request (8)
    Code: 0
    Checksum: 0x4d56 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 5 (0x0005)
    Sequence Number (LE): 1280 (0x0500)
    [Response frame: 971]
    Data (32 bytes)
        Data: 6162636465666768696a6b6c6d6e6f707172737475767761626364656667686 9
        [Length: 32]
```

```
0000  04 d4 c4 4f a9 d0 34 5a  60 db 19 6e 08 00 45 00
0010  00 3c 2f 74 00 00 80 01  00 00 c0 a8 32 40 08 08
0020  08 08 08 00 4d 56 00 01  00 05 61 62 63 64 65 66
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76
0040  77 61 62 63 64 65 66 67  68 69
```

icmp

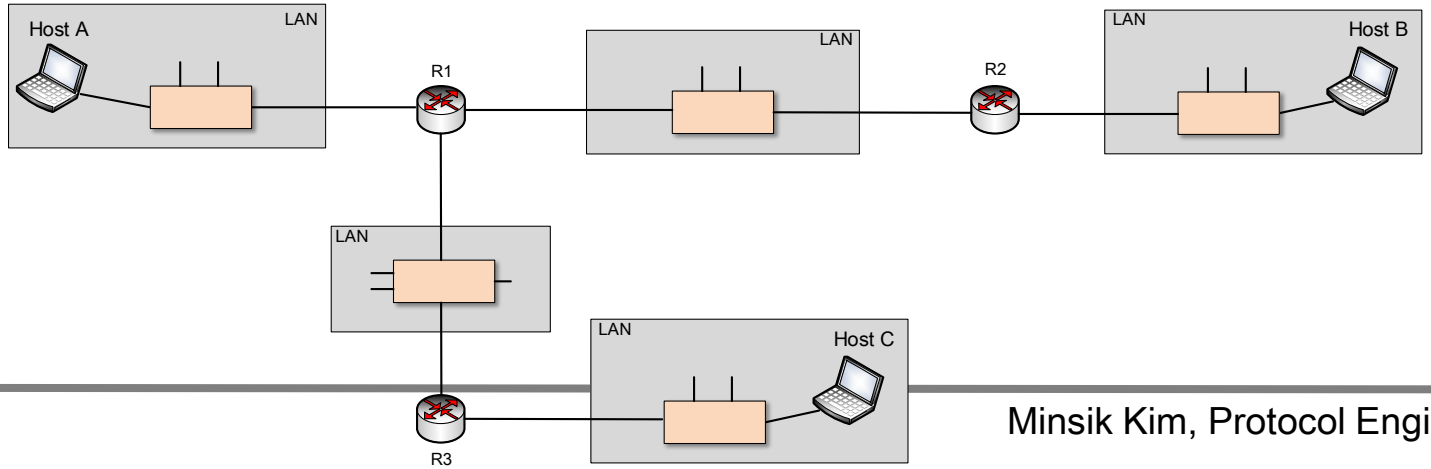
No.	Time	Source	Destination	Protocol	Lengt	Info
970	21.471569900	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=5/1280, ttl=128 (reply in 971)
971	21.629207500	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=5/1280, ttl=104 (request in 970)
975	22.479161200	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=6/1536, ttl=128 (reply in 977)
977	22.636272700	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=6/1536, ttl=104 (request in 975)
981	23.496511000	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=7/1792, ttl=128 (reply in 982)
982	23.653860000	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=7/1792, ttl=104 (request in 981)
1003	24.513238400	192.168.50.64	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=8/2048, ttl=128 (reply in 1005)
1005	24.670552000	8.8.8.8	192.168.50.64	ICMP	74	Echo (ping) reply id=0x0001, seq=8/2048, ttl=104 (request in 1003)

```
    Protocol: ICMP (1)
    Header Checksum: 0x4fa9 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 8.8.8.8
    Destination Address: 192.168.50.64
    [Stream index: 36]
Internet Control Message Protocol
    Type: Echo (ping) reply (0)
    Code: 0
    Checksum: 0x5556 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 5 (0x0005)
    Sequence Number (LE): 1280 (0x0500)
    [Request frame: 970]
    [Response time: 157.638 ms]
    Data (32 bytes)
        Data: 6162636465666768696a6b6c6d6e6f707172737475767761626364656667686 9
        [Length: 32]
```

```
0000  34 5a 60 db 19 6e 04 d4  c4 4f a9 d0 08 00 45 20
0010  00 3c 00 00 00 00 68 01  4f a9 08 08 08 08 c0 a8
0020  32 40 00 00 55 56 00 01  00 05 61 62 63 64 65 66
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76
0040  77 61 62 63 64 65 66 67  68 69
```

ICMPv4

- 디버깅 도구 (3/4)
  - Traceroute (1/2)
    - 정의
      - 송신지에서 목적지까지 패킷이 거치는 라우터의 경로를 추적하기 위한 네트워크 진단 프로그램
    - 특징
      - IP 패킷의 TTL(Time To Live) 값을 1부터 순차적으로 증가시키며 패킷을 전송함
      - 패킷 전송 시각과 ICMP 메시지 수신 시각의 차이를 이용해 각 구간의 왕복 시간(RTT)을 측정할 수 있음



ICMPv4

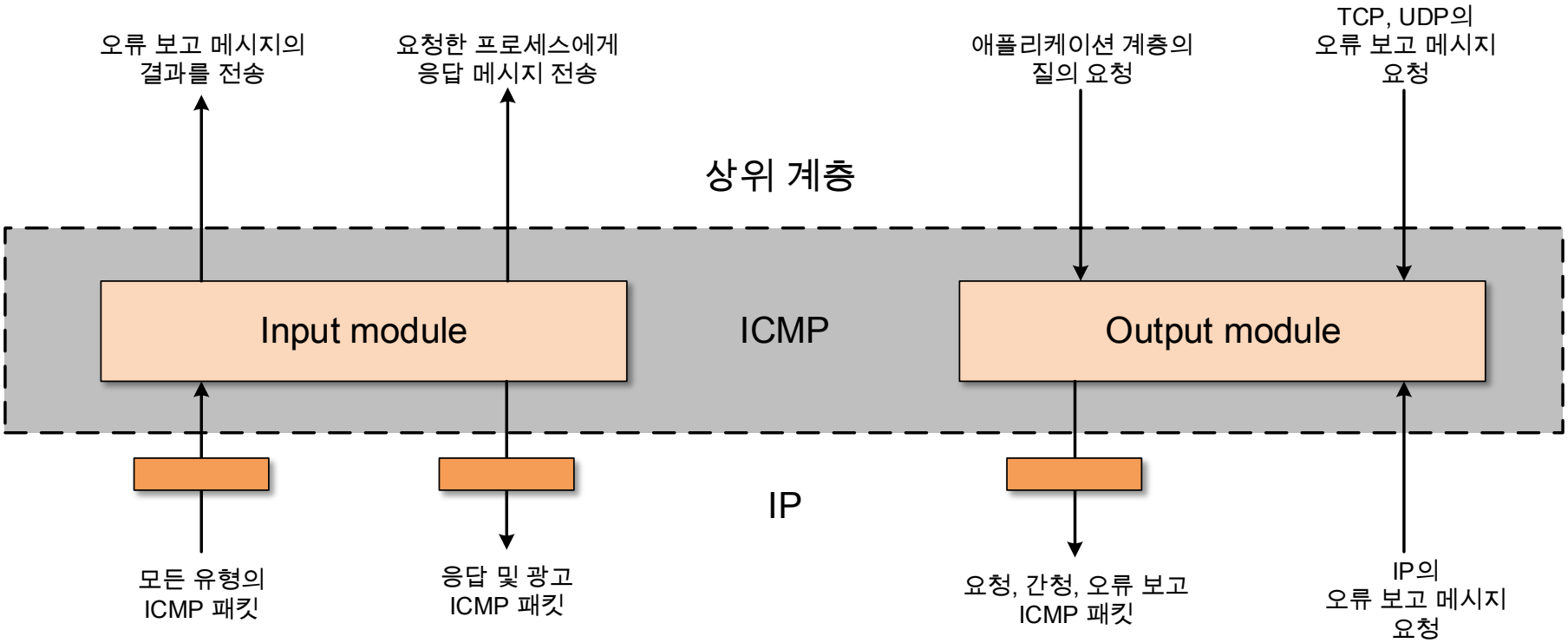
- 디버깅 도구 (4/4)
  - Traceroute (2/2)
    - ICMP 응답이 수신되지 않은 홉은 *로 표시되며, 이후 경로 추적은 계속될 수 있음

```
C:\Users\ms612> tracert 8.8.8.8

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

  1    <1 ms    <1 ms    <1 ms  router.asus.com [192.168.50.1]
  2     3 ms     2 ms     2 ms  223.195.38.1
  3     2 ms     1 ms     1 ms  210.107.239.14
  4     1 ms     1 ms     1 ms  210.107.239.9
  5     2 ms     2 ms     1 ms  210.107.239.1
  6     2 ms     2 ms     2 ms  211.168.5.113
  7     2 ms     2 ms     2 ms  1.213.141.9
  8     3 ms     2 ms     2 ms  1.208.107.189
  9     *        *        *     요청 시간이 만료되었습니다.
 10     3 ms     3 ms     3 ms  100.66.51.25
 11    37 ms    37 ms    37 ms  100.67.30.2
 12    40 ms    38 ms    38 ms  1.208.178.26
 13   167 ms   167 ms   167 ms  142.250.168.244
 14   164 ms   164 ms   164 ms  64.233.175.107
 15   161 ms   160 ms   168 ms  142.251.244.223
 16   157 ms   158 ms   157 ms  dns.google [8.8.8.8]

추적을 완료했습니다.
```

```
C:\Users\ms612>curl ipinfo.io/223.195.38.1
{
  "ip": "223.195.38.1",
  "city": "Sejong",
  "region": "Sejong-si",
  "country": "KR",
  "loc": "36.5924,127.2922",
  "org": "AS9769 Sejong University",
  "postal": "30020",
  "timezone": "Asia/Seoul",
  "readme": "https://ipinfo.io/missingauth"
}
```

ICMPv4

- ICMP 패키지 (1/3)
  - ICMP 메시지가 어떻게 송신되고 수신되어 처리되는지를 설명하기 위한 소프트웨어 구성
  - 입력 모듈은 수신된 ICMP 메시지를 처리하고, 출력 모듈은 요청에 따라 ICMP 메시지를 생성·전송함



ICMPv4

- ICMP 패키지 (2/3)
  - 입력 모듈
    - IP 계층으로부터 전달된 모든 수신 ICMP 메시지를 처리함
    - 수신된 패킷이 요청 메시지이면 모듈은 응답 메시지를 생성하여 송신함
    - 재지정 메시지면 메시지의 정보를 이용해 라우팅 테이블을 수정함

ICMPv4

- ICMP 패키지 (3/3)
  - 출력 모듈
    - 상위 계층이나 IP 프로토콜로부터 요구를 받아 ICMP 메시지를 생성함
    - IP, UDP, TCP로부터 오류 메시지 전송 요구를 받으면 해당 ICMP 오류 메시지를 생성함
    - 응용 프로그램으로부터 요청 메시지 전송 요구를 받으면 해당 ICMP 요청 메시지를 생성하여 송신함

목 차

- ARP(Address Resolution Protocol)
- ICMPv4(Internet Control Message Protocol Version 4)
- 모바일 IP(Mobile IP)

모바일 IP

- 주소 지정 (1/2)
  - 정지 호스트(Stationary Hosts)
    - 하나의 특정 네트워크에 계속 연결되어 있는 호스트
    - IP 주소의 프리픽스(Prefix)를 통해 호스트가 속한 네트워크를 식별함
  - 이동 호스트(Mobile Hosts)
    - 한 네트워크에서 다른 네트워크로 이동하는 호스트
    - 호스트가 한 네트워크에서 다른 네트워크로 이동하면 IP 주소 구조도 변경되어야 함

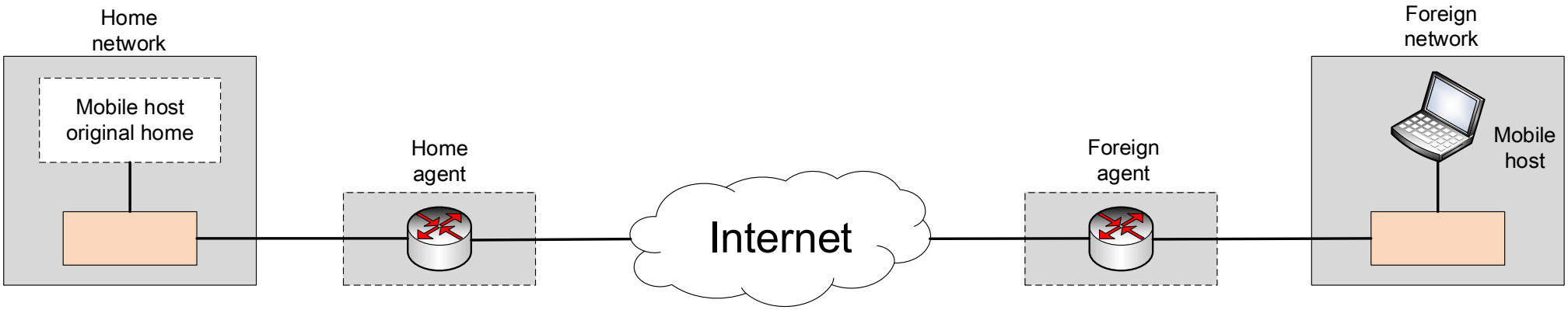
모바일 IP

- 주소 지정 (2/2)
  - 이동 호스트 주소 지정 방법
    - 주소 변경
      - 새로운 네트워크로 이동할 때 DHCP를 이용해 새로운 IP 주소를 할당받는 방식
      - 네트워크 이동 시 설정 파일과 DNS 테이블을 수정하고 재부팅해야 하며, 통신 중 주소가 변경되면 데이터 전송이 중단될 수 있음
    - 두 개의 주소
      - 이동 호스트가 영구적인 홈 주소(Home Address)와 의탁 주소 (Care-of Address) 두 개를 사용하는 방식
      - 홈 주소는 이동해도 유지되며, 의탁 주소는 이동한 네트워크에 따라 변경됨

*DHCP(Dynamic Host Configuration Protocol): 네트워크에 접속한 장치에 IP 주소 등의 네트워크 설정 정보를 자동으로 할당하는 프로토콜

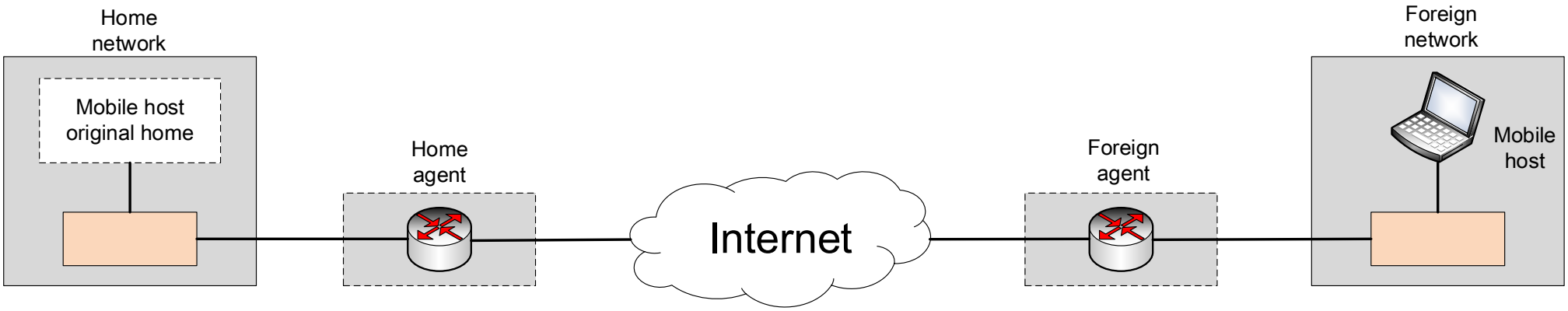
모바일 IP

- 에이전트(Agent) (1/3)
  - 정의
    - 이동 호스트의 이동성을 지원하기 위한 기능을 수행하는 라우터
  - 종류
    - 홈 에이전트(Home Agent)
    - 외부 에이전트(Foreign Agent)



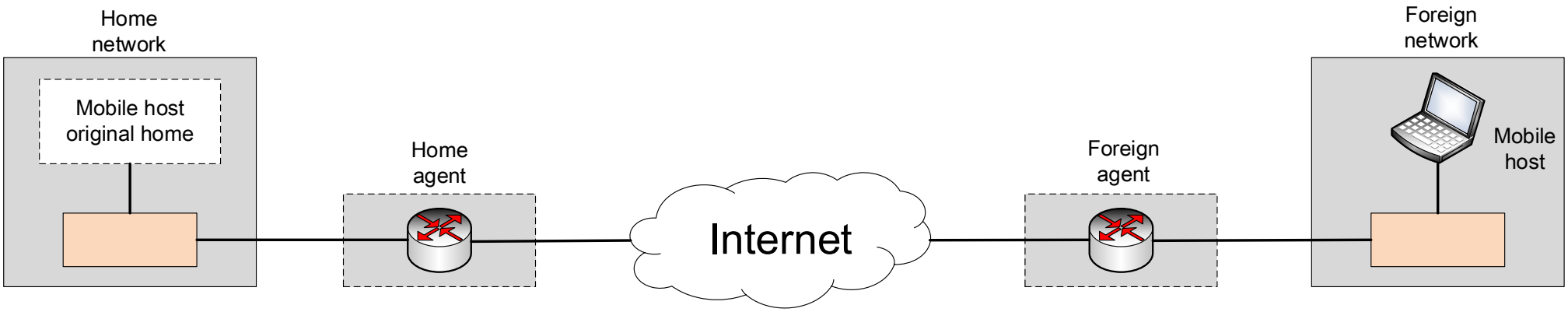
모바일 IP

- 에이전트(Agent) (2/3)
  - 홈 에이전트(Home Agent)
    - 이동 호스트의 홈 네트워크에 연결되어 이동 호스트를 대신해 패킷을 처리하는 에이전트
    - 원격 호스트가 이동 호스트로 보낸 패킷을 수신하여 외부 에이전트로 전달함



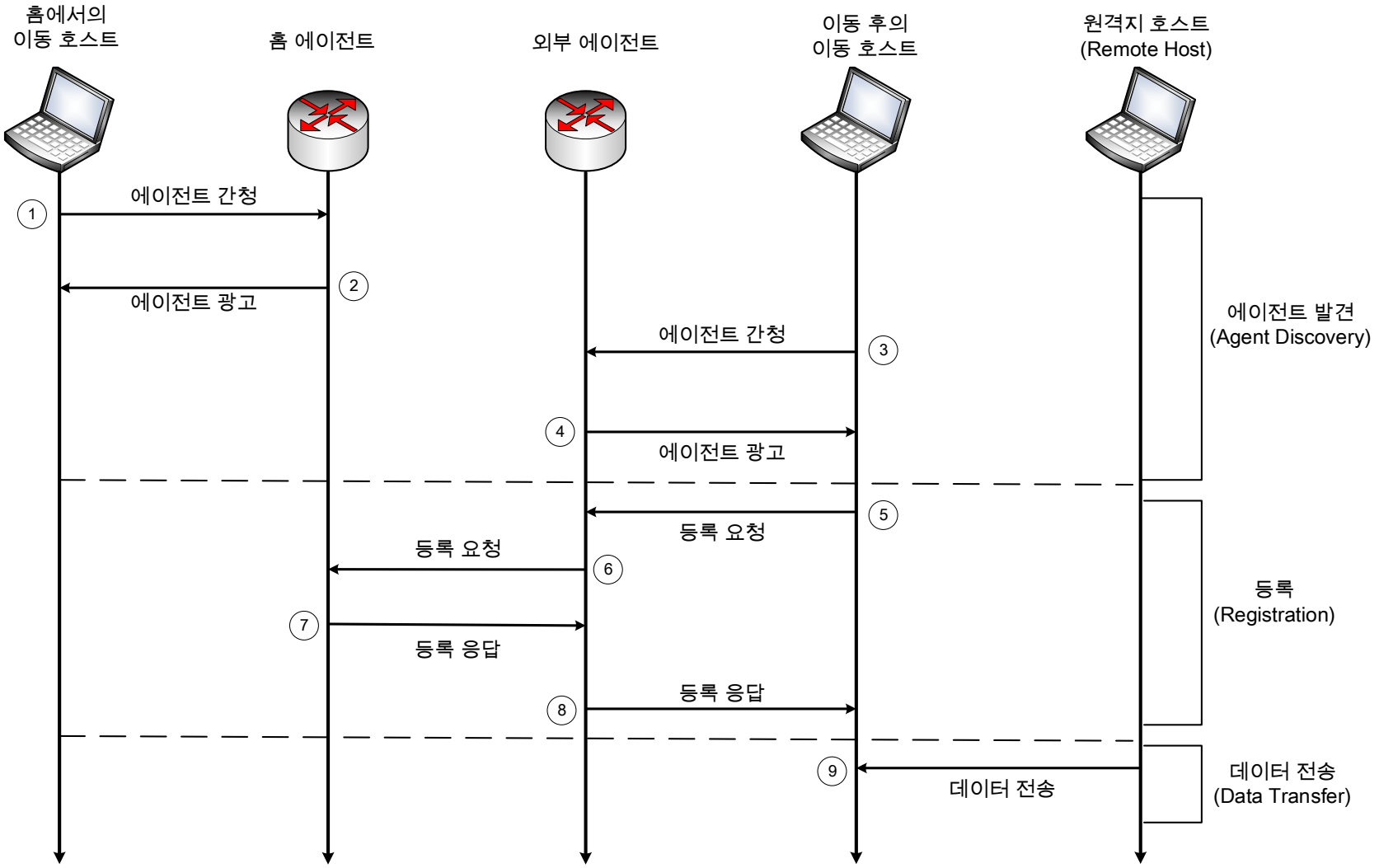
모바일 IP

- 에이전트(Agent) (3/3)
  - 외부 에이전트(Foreign Agent)
    - 이동 호스트가 방문한 외부 네트워크에 연결되어 이동 호스트의 패킷 전달을 지원하는 에이전트
    - 홈 에이전트가 보낸 패킷을 수신하여 이동 호스트에게 전달함



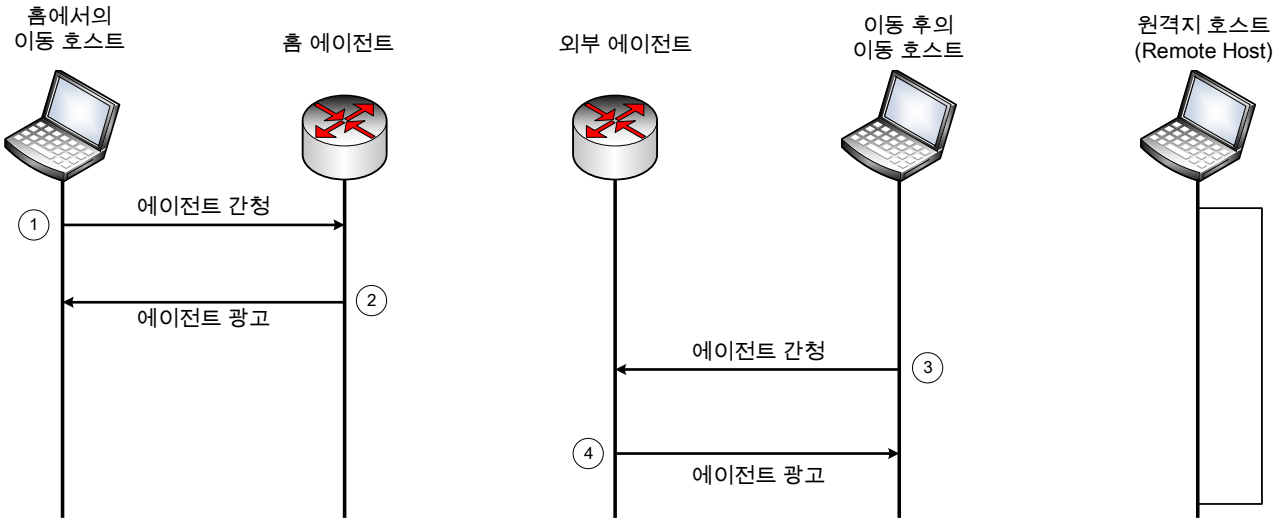
모바일 IP

- 과정 (1/9)


홈에서의
이동 호스트
홈 에이전트
외부 에이전트
이동 후의
이동 호스트
원격지 호스트
(Remote Host)
1
에이전트 간청
2
에이전트 광고
에이전트 간청
3
4
에이전트 광고
에이전트 발견
(Agent Discovery)
5
등록 요청
등록 요청
6
7
등록 응답
등록 응답
8
등록
(Registration)
9
데이터 전송
데이터 전송
(Data Transfer)

모바일 IP

- 과정 (2/9)
  - 에이전트 발견(Agent Discovery)
    - 이동 호스트가 통신에 필요한 홈 에이전트와 외부 에이전트의 정보를 확인하는 과정
    - 이동 호스트는 홈 네트워크를 떠나기 전에 홈 에이전트를 발견하고, 외부 네트워크로 이동한 후에는 외부 에이전트와 의탁 주소(Care-of Address)를 발견함



모바일 IP

- 과정 (3/9)
  - 에이전트 발견 – 광고(Advertisement)
    - 라우터가 네트워크에서 자신의 존재를 알릴 때, 에이전트로 동작하는 경우 ICMP Router Advertisement에 에이전트 광고 메시지를 추가하여 전송함
    - 메시지는 유형, 길이, 순서 번호, 유효 시간, 코드, 임시 주소 목록 필드로 구성됨

<table>
<tr><td colspan="4">ICMP
Advertisement message</td></tr>
<tr><td>Type</td><td>Type</td><td colspan="2">Sequence number</td></tr>
<tr><td colspan="2">Lifetime</td><td>Code</td><td>Reserved</td></tr>
<tr><td colspan="4">Care-of addresses
(foreign agent only)</td></tr>
</table>

모바일 IP

- 과정 (4/9)
  - 에이전트 발견 – 간청(Solicitation)
    - 이동 호스트가 새로운 네트워크로 이동한 후 에이전트 광고를 받지 못했을 때 에이전트에게 도움을 요청하기 위해 에이전트 간청 메시지를 전송함
    - 별도의 새로운 패킷을 사용하지 않고 ICMP 라우터 간청 (Router Solicitation) 메시지를 이용함

Type: 10	Code: 0	Checksum
Reserved		

모바일 IP

- 과정 (5/9)
  - 등록(Registration)
    - 이동 호스트는 외부 에이전트에 등록 요청을 전송함
    - 외부 에이전트는 해당 등록 요청을 홈 에이전트에 전달함
    - 등록의 유효 기간이 만료되면 등록을 갱신해야 함
    - 이동 호스트가 홈 네트워크로 돌아오면 등록을 해제 (Deregistration)해야 함

모바일 IP

- 과정 (6/9)
  - 등록 – 요청(Request)
    - 이동 호스트가 의탁 주소를 등록하고 홈 주소와 홈 에이전트 주소를 알리기 위해 외부 에이전트로 등록 요청을 전송함
    - 외부 에이전트는 등록 요청을 받은 후 홈 에이전트로 전달함

Type: 1	Flag	Lifetime
Home address		
Home agent address		
Care-of address		
Identification		
Extensions		

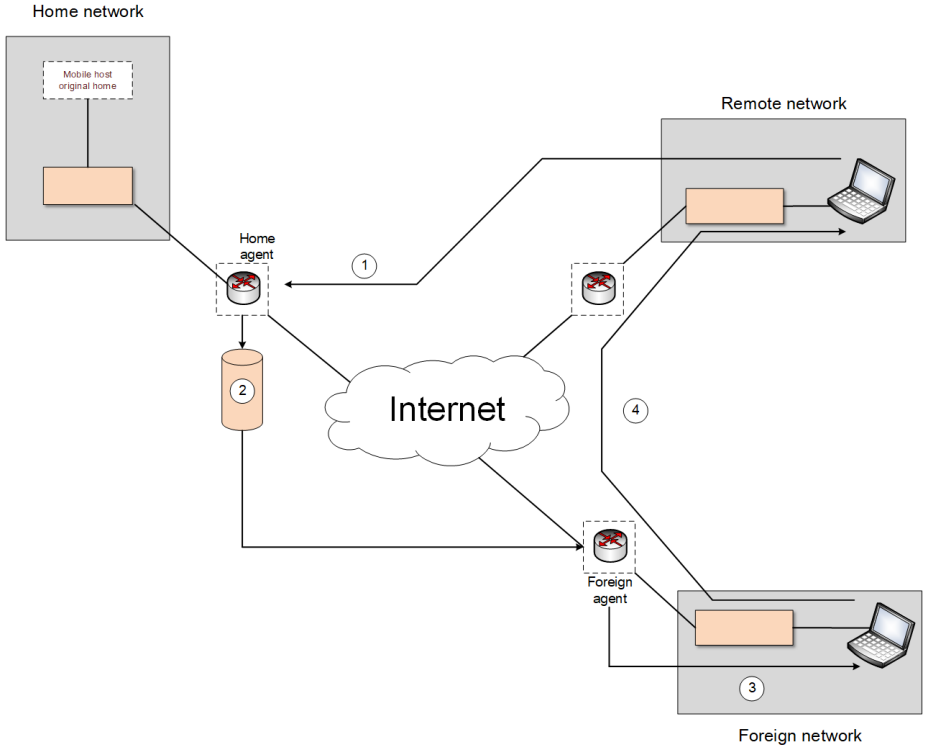
모바일 IP

- 과정 (7/9)
  - 등록 – 응답(Response)
    - 홈 에이전트가 등록 요청의 승인 또는 거부 결과를 외부 에이전트로 전송하고, 외부 에이전트가 이를 이동 호스트로 전달함
    - 유형(Type)은 3이며, 코드(Code) 필드를 통해 등록 요청의 승인 또는 거부 결과를 나타냄

<table>
<tr><td>Type: 3</td><td>Flag</td><td>Lifetime</td></tr>
<tr><td colspan="3">Home address</td></tr>
<tr><td colspan="3">Home agent address</td></tr>
<tr><td colspan="3">Identification</td></tr>
<tr><td colspan="3">Extensions</td></tr>
</table>

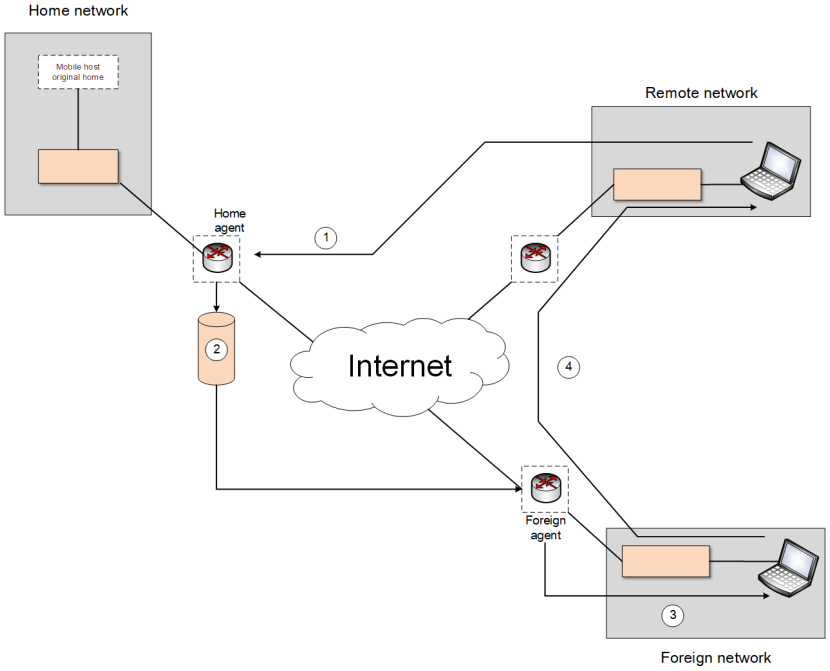
모바일 IP

- 과정 (8/9)
  - 데이터 전송(Data Transfer) (1/2)
    1. 원격지 호스트가 이동 호스트의 홈 주소로 패킷을 전송하면 홈 에이전트가 해당 패킷을 가로챔
    2. 홈 에이전트는 해당 패킷을 캡슐화하여 터널링을 통해 외부 에이전트로 전송함



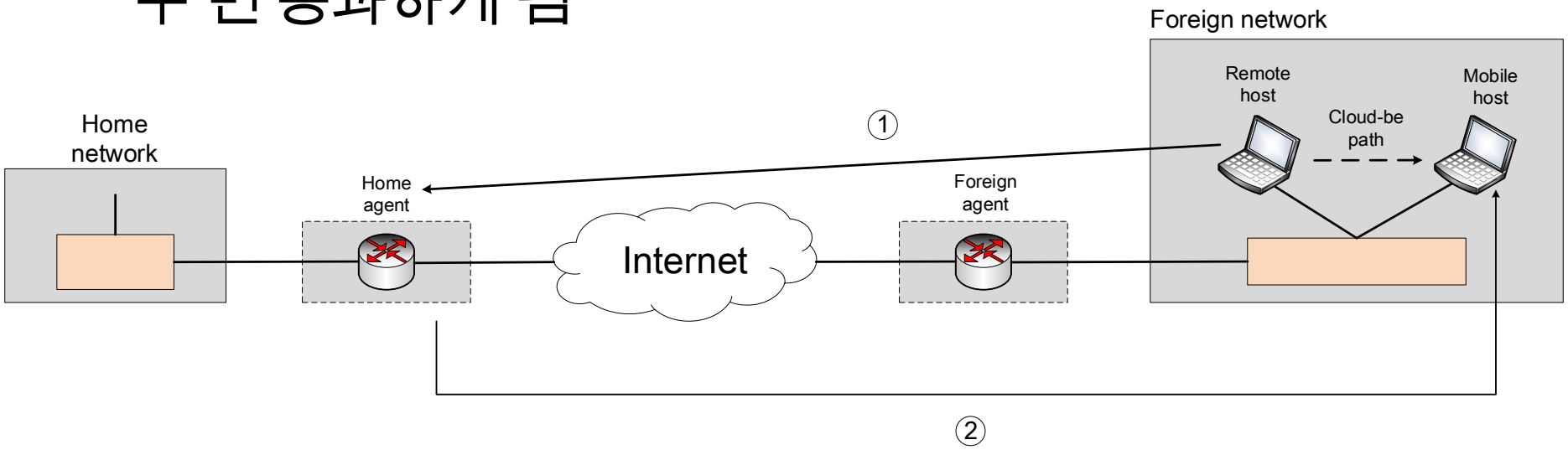
모바일 IP

- 과정 (9/9)
  - 데이터 전송(Data Transfer) (2/2)
    3. 외부 에이전트는 패킷을 역캡슐화한 후 이동 호스트의 의탁 주소를 확인하여 패킷을 전달함
    4. 이동 호스트는 자신의 홈 주소를 출발지 주소로 사용하여 원격지 호스트로 직접 패킷을 전송함



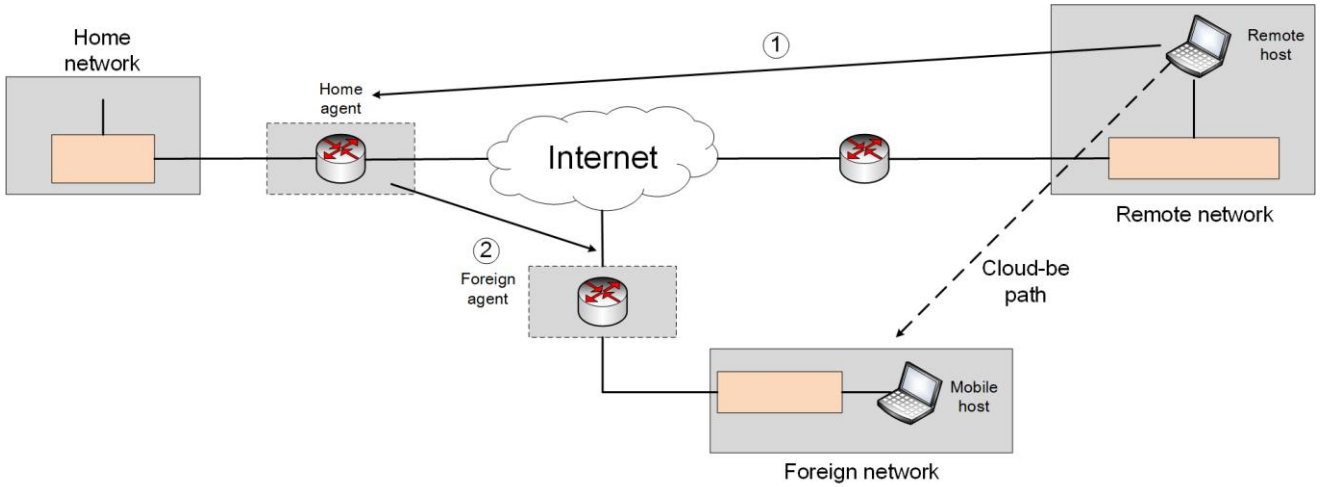
모바일 IP

- 비효율성 문제 (1/3)
  - 더블 크로싱(Double Crossing)
    - 원격지 호스트와 이동 호스트가 같은 네트워크에 있음에도, 원격지 호스트가 보낸 패킷이 홈 에이전트를 거쳐 다시 이동 호스트로 전달되는 비효율적인 경로
    - 이동 호스트 → 원격지 호스트는 로컬 통신이지만, 원격지 호스트 → 이동 호스트는 패킷이 인터넷을 두 번 통과하게 됨



모바일 IP

- 비효율성 문제 (2/3)
  - 삼각형 라우팅(Triangle Routing)
    - 원격지 호스트가 이동 호스트로 보낸 패킷이 홈 에이전트를 거쳐 이동 호스트로 전달되어 삼각형 형태의 비효율적인 경로가 형성되는 현상
    - 이동 호스트 → 원격지 호스트는 직접 전송되지만, 원격지 호스트 → 이동 호스트는 홈 에이전트를 경유하여 메시지를 전달하게 됨



모바일 IP

- 비효율성 문제 (3/3)

*바인딩:
두 주소를 서로 연결하여 대응 관계를 설정하는 것

  - 해결책(Solution)
    - 원격지 호스트가 이동 호스트의 홈 주소와 의탁 주소를 바인딩하여 이후 패킷을 의탁 주소로 직접 전송함
      - 홈 에이전트를 우회하여 더블 크로싱 및 삼각형 라우팅의 비효율적인 전달 경로 문제를 해결함
    - 홈 에이전트는 첫 패킷을 전달하면서 원격지 호스트에 바인딩 갱신 패킷을 전송하고, 원격지 호스트는 해당 정보를 캐시에 저장함
      - 원격지 호스트가 이동 호스트로 직접 전송할 수 있도록 최신 바인딩 정보를 제공함
    - 이동 호스트가 다른 네트워크로 이동해 캐시 정보가 오래되면 홈 에이전트가 경고 패킷을 보내 변경 사실을 알림
      - 오래된 바인딩 정보로 인한 잘못된 패킷 전달 문제를 방지함

Thanks!

김 민 식(minsik@pel.sejong.ac.kr)