

TCP/IP 프로토콜

-11장 유니캐스트 프로토콜,
12장 멀티캐스트 프로토콜-

민 윤 홍(yunhong@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

목 차

- 유니캐스트 라우팅 프로토콜
 - 도메인 내 및 도메인 간 라우팅
 - 거리 벡터 라우팅
 - 링크 상태 라우팅
 - 경로 벡터 라우팅
- 멀티캐스트 라우팅 프로토콜
 - 유니캐스팅과 멀티캐스팅
 - IGMP (Internet Group Management Protocol)
 - 멀티캐스트 라우팅

도메인 내 및 도메인 간 라우팅

- 자율 시스템(AS, Autonomous System)

- 정의

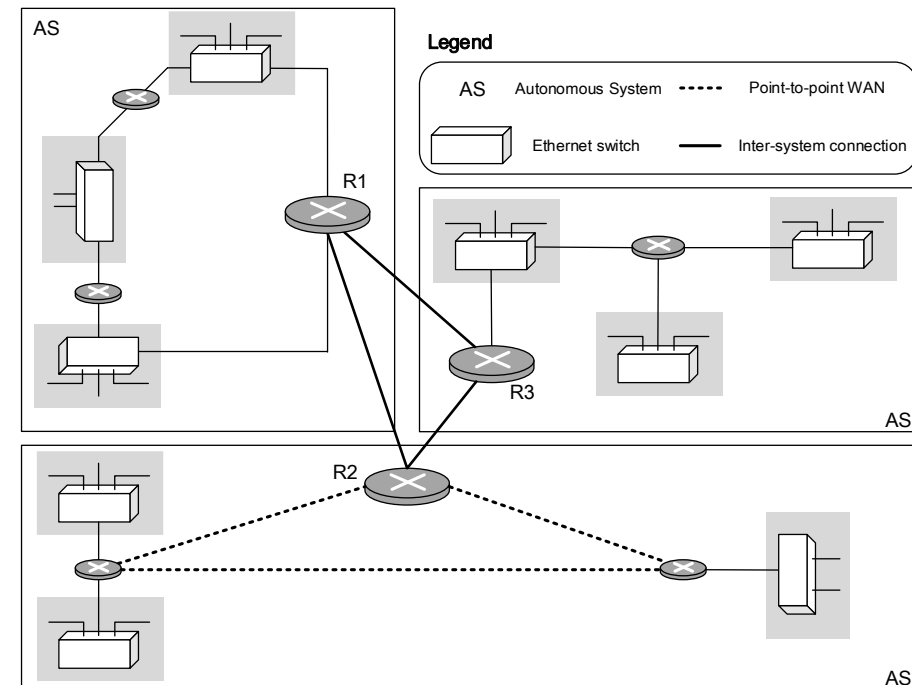
- 하나의 단일 관리 기관 하에 있는 라우터와 네트워크의 그룹
 - e.g., KT, SKT, Google 같이 하나의 회사가 관리하는 인터넷의 구역

- 도메인 내 라우팅(Intradomain Routing)

- 자율 시스템 내에서의 라우팅
 - 하나 이상의 도메인 내 라우팅 프로토콜을 사용

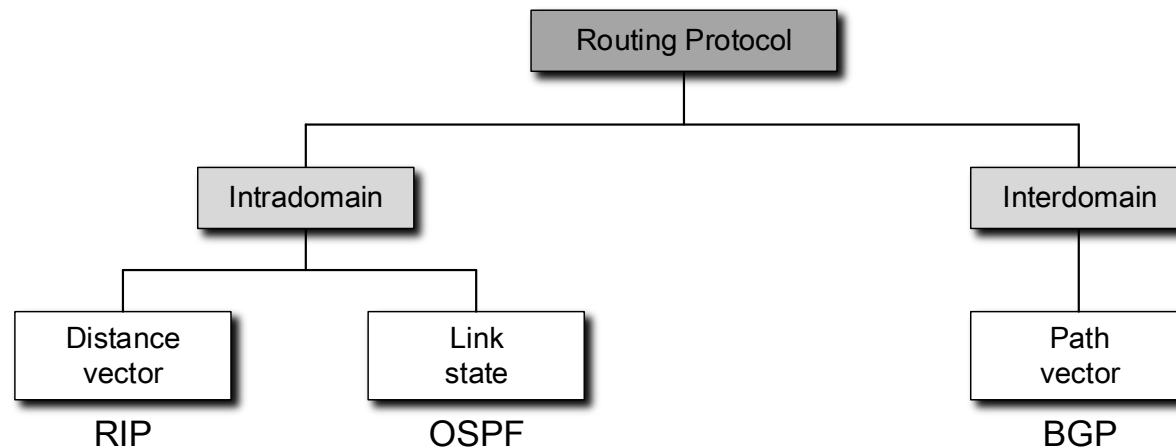
- 도메인 간 라우팅(Interdomain Routing)

- 자율 시스템 간의 라우팅
 - 하나의 도메인 간 하나의 라우팅 프로토콜을 사용



도메인 내 및 도메인 간 라우팅

- 자율 시스템(AS, Autonomous System)
 - 라우팅 프로토콜
 - 정의
 - 인터넷상의 라우터가 각각의 변화를 서로에게 알릴 수 있도록 하기 위한 규칙과 절차의 조합
 - 종류
 - RIP (Routing Information Protocol)
 - OSPF (Open Shortest Path First)
 - BGP (Border Gateway Protocol)



목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

거리 벡터 라우팅

• 정의

- 각 라우터가 자신의 *라우팅 테이블을 인접한 라우터와 주기적으로 경로 정보를 교환하고, 이를 자신의 경로 정보와 비교하여 최적 경로를 결정하는 방식

• 특징

- 분산형 정보 관리를 통해 라우터의 부담 완화
- RIP (Routing Information Protocol) 로 구현
- Bellman-Ford 변형 알고리즘을 사용

IPv4 경로 테이블

형성 경로:
네트워크 대상 네트워크 마스크 게이트웨이 인터페이스 메트릭

루프백 네트워크 (내 PC 안에서만 쓰는 가상 인터페이스)	가상/별도 어댑터 (VirtualBox)	브로드캐스트 주소	서브넷 라우트 서브넷 브로드캐스트 주소	인터넷 공유기 (라우터)	자신의 IP 주소	메트릭
127.0.0.0	0.0.0.0	255.0.0.0		192.168.50.1	192.168.50.77	35
127.0.0.1	255.255.255.255					331
127.255.255.255	255.255.255.255					331
192.168.15.0	255.255.255.0					291
192.168.15.1	255.255.255.255					291
192.168.15.255	255.255.255.255					291
192.168.50.0	255.255.255.0					291
192.168.50.77	255.255.255.255					291
192.168.50.255	255.255.255.255					291
192.168.146.0	255.255.255.0					291
192.168.146.1	255.255.255.255					291
192.168.146.255	255.255.255.255					291

*라우팅 테이블(Routing table)
라우터가 네트워크 트래픽을 어디로 보내야 할지 결정하는 데 사용하는 경로 정보

거리 벡터 라우팅

- Bellman-Ford 알고리즘 (1/2)

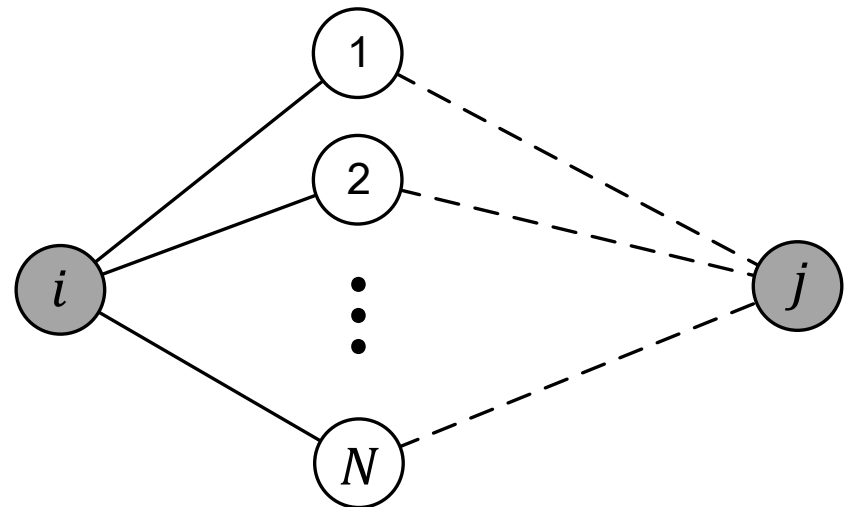
- 정의

- 임의의 두 노드 간의 비용을 안다면 두 노드 간의 최소 비용 (최단 거리)을 찾을 수 있는 알고리즘

$$D_{ij} = \text{minimum} \{ (c_{i1} + D_{1j}), (c_{i2} + D_{2j}) \cdots (c_{iN} + D_{Nj}) \}$$

Legend

D_{ij} Shortest distance between i and j
 c_{ij} Cost between i and j
 N Number of nodes



거리 벡터 라우팅

• Bellman-Ford 알고리즘 (2/2)

• 동작 과정

1. 각 노드와 자신 간의 최단 거리를 0으로 초기화
2. 각 노드와 다른 노드와의 최단 거리를 무한대로 설정
3. 노드와 다른 노드 간의 비용을 계산하여 설정
4. 만약 연결이 없다면 무한대로 유지
5. 최단 거리 벡터에 변경 사항이 없을 때까지 알고리즘 반복

```
1. Bellman_Ford ()
2. {
3.     // Initialization
4.     for (i = 1 to N; for j = 1 to N)
5.     {
6.         if (i == j) D[i][j] = 0
7.         c[i][j] = 0
8.         else D[i][j] = ∞
9.         c[i][j] = cost between i and j
10.    }
11.
12.    // Updating
13.    repeat
14.    {
15.        for (i = 1 to N; for j = 1 to N)
16.        {
17.            D[i][j] = min(c[i][1] + D[1][j], ..., c[i][N] + D[N][j])
18.        } // end for
19.    } until (there was no change in previous iteration)
20. } // end Bellman-Ford
```

거리 벡터 라우팅

• Bellman-Ford 변형 알고리즘 (1/11)

*홉 수(Hop count)

목적지에 도달하기 위해 거쳐 가야 하는 노드의 수

• 개요

- 중앙에서 한 번에 계산하는 Bellman-Ford 알고리즘을 분산형인 거리 벡터 라우팅에 사용하기 위해 변형 필요

• 동작 과정 (1/2)

1. 비용은 *홉(Hop)수를 사용, 연결된 두 노드 간의 비용은 1로 설정
2. 각 라우터는 Bellman-Ford 알고리즘의 일부를 수행
 - 더 좋은 경로를 발견하게 되면 기존 경로를 수행하는 부분을 수행
 - 기존 Bellman-Ford 알고리즘은 이웃 노드가 아니라, 하나의 시스템이 전체 네트워크에 해당하는 모든 노드와 연결 비용을 알아야 함
3. 각 라우터가 자신의 라우팅 테이블을 갱신한 후 그 결과를 이웃 노드에게 전달하여 그들이 테이블을 갱신할 수 있도록 함

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (2/11)

- 동작 과정 (2/2)

4. 라우터는 목적지망 비용, 다음 홉에 대한 정보를 포함
 - 전체 라우팅 테이블은 $Table$, 테이블의 i 번째 행은 $Table_i$, i 번째에 해당하는 세 가지 열 정보는 $Table_i.dest$, $Table_i.cost$, $Table_i.next$
5. 이웃 노드로부터 받은 각 경로에 대한 레코드 정보 R 은 $R.dest$, $R.cost$ 를 포함

표기법	의미
$Table$	전체 라우팅 테이블
$Table_i$	테이블의 i 번째 행
$Table_i.dest$	목적지 노드
$Table_i.cost$	현재 라우터에서 목적지까지의 비용
$Table_i.next$	목적지로 가기 위해 선택된 다음 홉 라우터
$R.dest$	이웃 노드가 광고한 목적지 노드
$R.cost$	이웃 노드가 광고한 목적지까지의 비용

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (3/11)

- 취약점

- 알고리즘 자체에는 별도의 검증이나 인증 과정이 없어 제3자에 의한 테이블 변조 공격에 취약함
 - 더 효율적인 경로를 선택하는 방식을 악용한 공격에 취약함
 - Black hole 공격
 - 악성 라우터가 자신을 거치면 목적지까지 비용이 1이라고 광고한 후, 다른 라우터들이 악성 라우터에 트래픽을 보내면 악성 라우터는 모든 트래픽을 버려 통신을 마비시킴
 - 라우팅 루프 유도
 - 악성 라우터가 자신을 거치면 목적지까지 비용이 1이라고 광고한 후, 의도적으로 무한 루프를 만들어 목적지로 패킷을 보낼 수 없게 함

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (4/11)
 - 해결 방안 (1/2)
 - 루프 방지 메커니즘 추가
 - 무한대 범위 제한
 - 무한대를 16과 같은 수로 재정의하여 최대 홉 수가 15 홉을 넘을 수 없게 하는 방법
 - 해당 알고리즘을 사용할 시기에는 소규모의 네트워크를 대상으로 설계되서 15를 넘는 경로는 사용할 일이 거의 없었음
 - RFC 1058 문서에서 최대 홉 수가 16일 경우 도달할 수 없는 목적지라고 판단

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (5/11)
 - 해결 방안 (2/2)
 - 인증 (Authentication) 기능 추가
 - 같은 네트워크의 라우터끼리 미리 약속된 비밀키를 공유
 - 해당 라우터가 라우팅 정보를 보낼 때 비밀키를 이용해 메시지를 암호화해서 보냄
 - 정보를 받은 라우터는 비밀키로 동일하게 암호화를 진행함
 - 암호화한 값이 서로 일치하면 라우팅 테이블에 반영함
 - RIPv2 에서 인증 기능을 추가하여 보완함

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (6/11)
- 알고리즘 가상 코드 (1/6)

```
1. Distance_Vector_Algorithm ()
2. {
3.     // At startup
4.     for (i = 1 to N)          // N is number of ports
5.     {
6.         Table[i].dest = address of the attached network
7.         Table[i].cost = 1
8.         Table[i].next = -    // Means at home
9.         Send(R[i])          // Send a record R about each row to each neighbor
10.    } // end for loop
11. }
```

- 라우터는 각 인터페이스에 직접 연결된 망들로 패킷을 라우팅하는데 사용되는 초기 라우팅 테이블을 생성함
- 라우팅 테이블에 있는 각 항목에 대해 목적지 망과 이웃 노드로의 비용을 전송함
- 두 노드 간 비용을 1로 설정하고, 관련 변수 Table, R을 사용함

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (7/11)
- 알고리즘 가상 코드 (2/6)

```
12. // Updating
13.   repeat (forever)
14.   {
15.       wait for arrival of a record R from a neighbor
16.       Update (R, Table)           // Call update module
17.       for (i = 1 to N)           // N is the current table size
18.       {
19.           Send(R[i])              // Send a record R about each row to each
neighbor
20.       }
21.   } // end repeat
22. } // end Distance_Vector
```

- 각 라우터는 이웃으로부터 레코드 정보(R)를 받으면 자신의 라우팅 테이블을 갱신함
- 자신의 테이블을 갱신한 결과를 이웃 노드에게 전달함

거리 벡터 라우팅

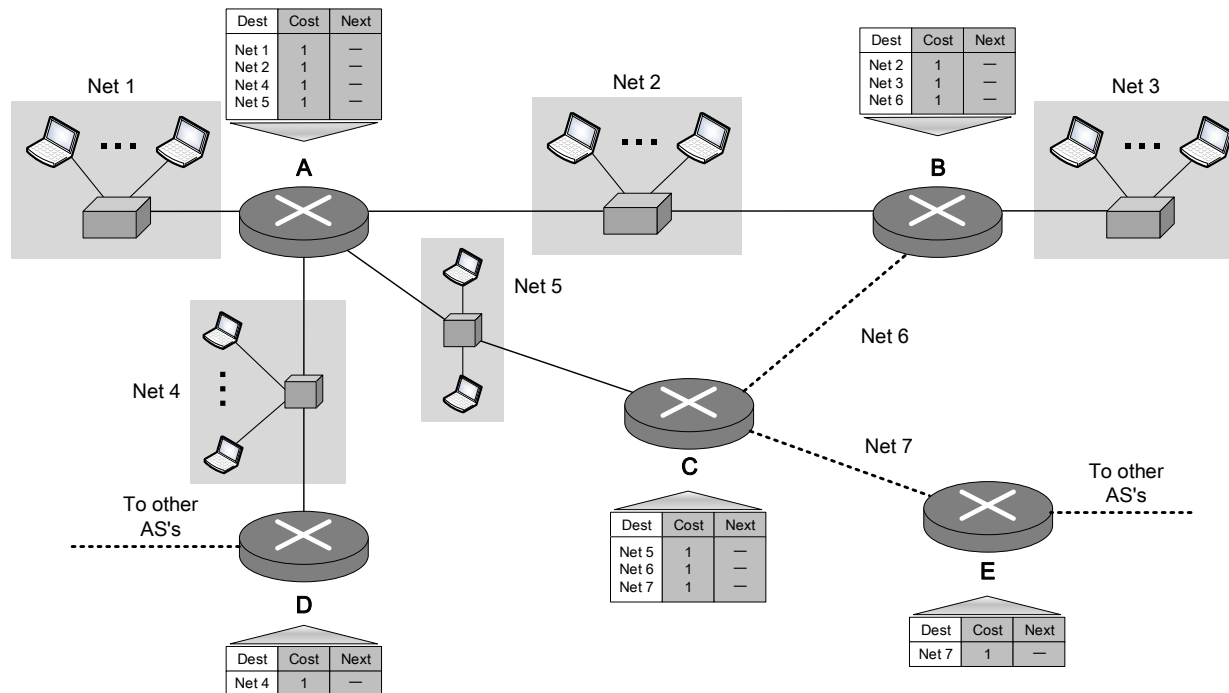
- Bellman-Ford 변형 알고리즘 (8/11)
- 알고리즘 가상 코드 (3/6)
 - 레코드가 도착하면, 라우터는 라우팅 테이블에 있는 목적지 주소를 검색함

```
24. Update (R, Table)
25. {
26.     Search Table for a destination matching the one in R
27.     if (destination is found in row i)
28.     {
29.         if (R.cost + 1 <= Table[i].cost || R.next == Table[i].next)
30.         {
31.             Table[i].cost = R.cost + 1
32.             Table[i].next = Address of sending router
33.         }
34.         else discard the record    // No change is needed
35.     }
36.     else
37.     {
38.         // Insert the new router
39.         Table[N+1].dest = R.dest
40.         Table[N+1].cost = R.cost + 1
41.         Table[N+1].next = Address of sending router
42.         Sort(Table, orderby=destination_address)
43.         // Sort the table according to destination address
44.     }
45. } // end of Update module
```

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (9/11)
- 알고리즘 가상 코드 (4/6)
 - 예제 11.1

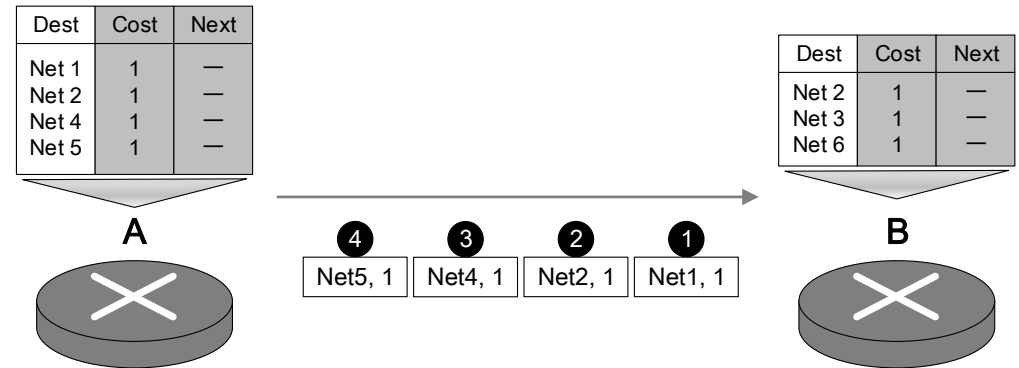
다음과 같은 네트워크에서 라우터 A가 이웃 노드인 라우터 B에 Net1, Net2, Net4, Net5 항목을 전송할 때, 라우터 B에서 라우팅 테이블의 변화를 보이시오.



거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (10/11)

- 알고리즘 가상 코드 (5/6)
 - 예제 11.1, 11.2



- 풀이 (1/2)

- 라우터 B가 레코드 1을 수신하면, Net1으로 가는 경로를 라우팅 테이블에서 검색하고 찾을 수 없기 때문에 비용을 1 더해서 (A와 B 사이의 거리) 기록하게 됨
- 레코드 2를 수신하면 경로가 있음을 확인하고, 비용을 1 더해서 테이블에 있던 값과 비교, 테이블에 있는 값보다 크기 때문에 해당 레코드를 버림

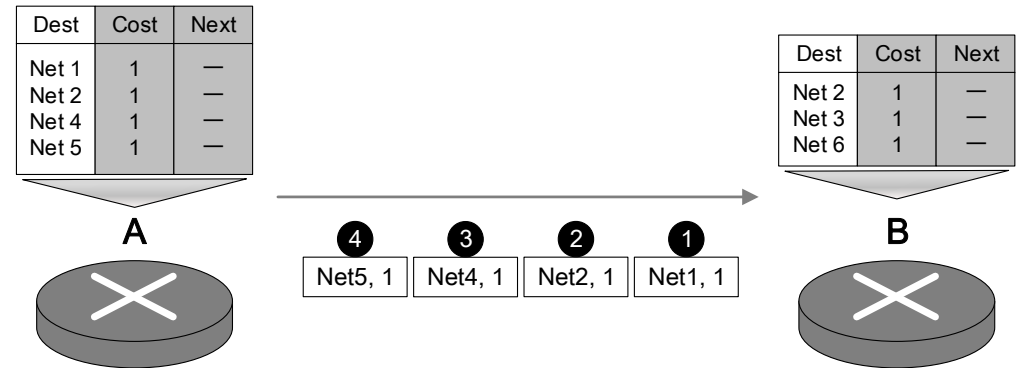
Routing Table B			Routing Table B			Routing Table B			Routing Table B		
Dest	Cost	Next	Dest	Cost	Next	Dest	Cost	Next	Dest	Cost	Next
Net 1	2	A	Net 1	2	A	Net 1	2	A	Net 1	2	A
Net 2	1	—	Net 2	1	—	Net 2	1	—	Net 2	1	—
Net 3	1	—	Net 3	1	—	Net 3	1	—	Net 3	1	—
Net 6	1	—	Net 6	1	—	Net 4	2	A	Net 4	2	A
						Net 6	1	—	Net 5	2	A
									Net 6	1	—
After receiving recode 1			After receiving recode 2			After receiving recode 3			After receiving recode 4		

거리 벡터 라우팅

- Bellman-Ford 변형 알고리즘 (11/11)

- 알고리즘 가상 코드 (6/6)

- 예제 11.1, 11.2



- 풀이 (2/2)

- 라우터 B가 레코드 3을 수신할 때, 테이블에 Net4가 없기 때문에 비용에 1을 더하여 테이블에 추가됨
- 라우터 B가 레코드 4를 수신할 때, 테이블에 Net5가 없기 때문에 비용에 1을 더하여 테이블에 추가됨

Routing Table B			Routing Table B			Routing Table B			Routing Table B		
Dest	Cost	Next	Dest	Cost	Next	Dest	Cost	Next	Dest	Cost	Next
Net 1	2	A	Net 1	2	A	Net 1	2	A	Net 1	2	A
Net 2	1	—	Net 2	1	—	Net 2	1	—	Net 2	1	—
Net 3	1	—	Net 3	1	—	Net 3	1	—	Net 3	1	—
Net 6	1	—	Net 6	1	—	Net 4	2	A	Net 4	2	A
						Net 6	1	—	Net 5	2	A
									Net 6	1	—
After receiving recode 1			After receiving recode 2			After receiving recode 3			After receiving recode 4		

거리 벡터 라우팅

- 무한대로의 카운트(Count to Infinity) 문제 (1/6)

- 정의

- 링크의 오류 등으로 인해 비용이 무한대로 바뀌었을 때, 이를 모든 라우터가 인식하는데 시간이 소요되어 발생하는 문제

- 발생 원인

- 거리 벡터 라우팅에서 비용 감소는 빠르게 확산되지만 비용 증가와 같은 정보는 느리게 확산되기 때문에 발생
 - 값이 줄어드는 계산은 즉시 갱신되고 주변에 알리는 반면, 값이 증가하는 계산은 기존 정보를 유지하려는 경향이 있음

거리 벡터 라우팅

• 무한대로의 카운트(Count to Infinity) 문제 (2/6)

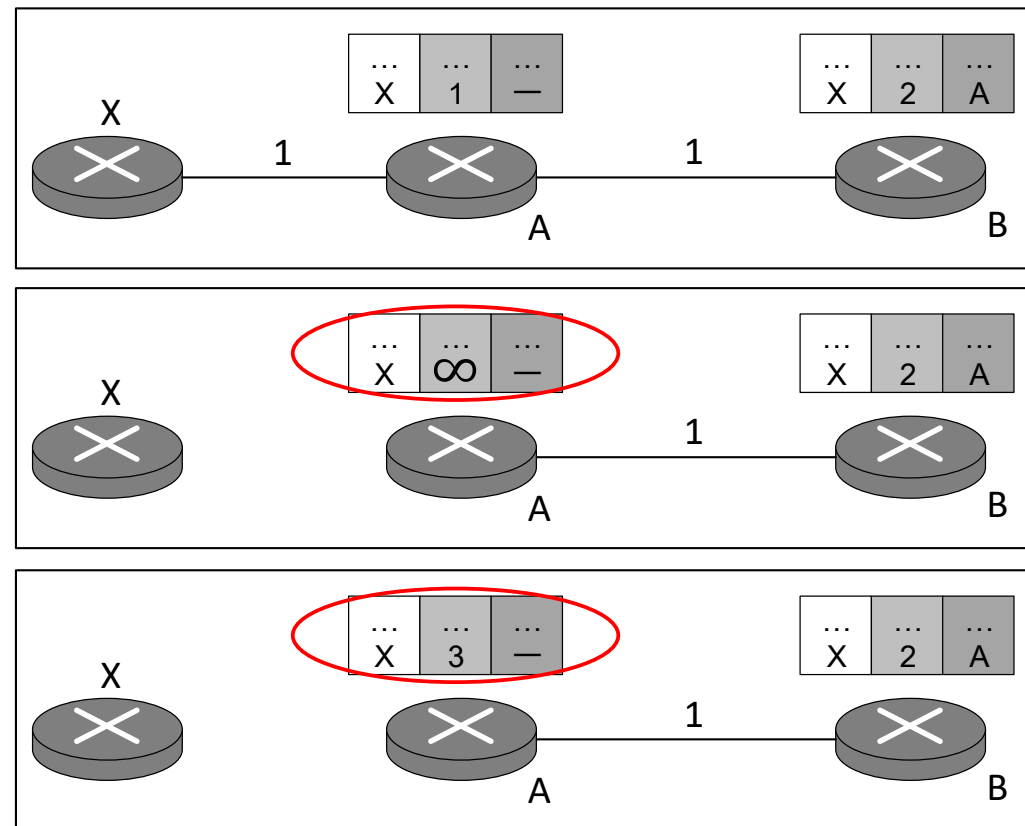
• 두 노드 루프(1/3)

• 발생 과정

1. A와 B는 노드 X에 도달하는 방법을 알고 있음

2. A와 X 사이의 링크가 실패하여, A는 자신의 테이블을 변경함

3. B가 A의 라우팅 테이블 정보를 받기 전에, 자신의 라우팅 테이블을 전송



거리 벡터 라우팅

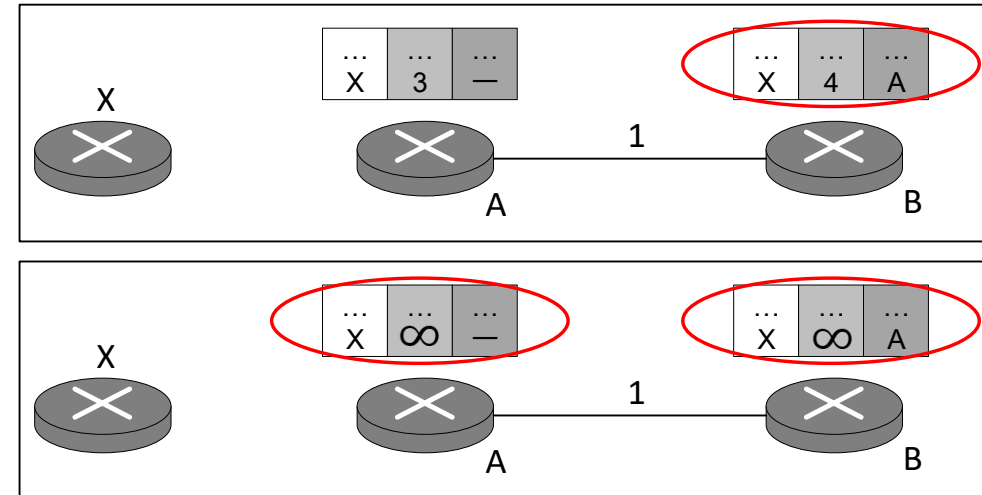
- 무한대로의 카운트(Count to Infinity) 문제 (3/6)

- 두 노드 루프(2/3)

- 발생 과정

4. A는 B의 라우팅 테이블 정보를 받고 자신의 라우팅 테이블을 갱신한 내용을 B에게 보냄

5. 위 과정을 반복하여, 비용이 무한대가 되어야 X에 도달할 수 없음을 알게 됨



거리 벡터 라우팅

• 무한대로의 카운트(Count to Infinity) 문제 (4/6)

• 두 노드 루프(3/3)

• 해결 방안

• 무한대 범위 제한

- 무한대를 16과 같은 수로 재정의하여 망의 크기가 15 홉을 넘을 수 없게 하는 방법

• 수평 분할(Split Horizon)

- 인터페이스를 통해 테이블을 *플러딩 하는 대신 각 인터페이스를 통해 자신 테이블의 일부만을 전송

- e.g., B가 X에 도달하는 최적의 경로가 A를 거치는 것이라는 걸 알고 A에게 다시 알리지 않음

• 포이즌 리버스(Poison Reverse)

- 이웃 라우터들에게 자신이 경유지로 사용되는 경로의 거리를 무한대로 알림

- e.g., B가 X에 대해 정보를 보낼 때 송신자가 A인 경우 거리 값을 무한대로 설정

- 수평 분할 정책과 조합해서 사용

*플러딩(Flooding)

라우터나 스위치가 들어온 데이터를 어디로 보내야 할 지 모를 때, 받은 포트를 제외한 모든 포트로 복사해서 보내는 전파 방식

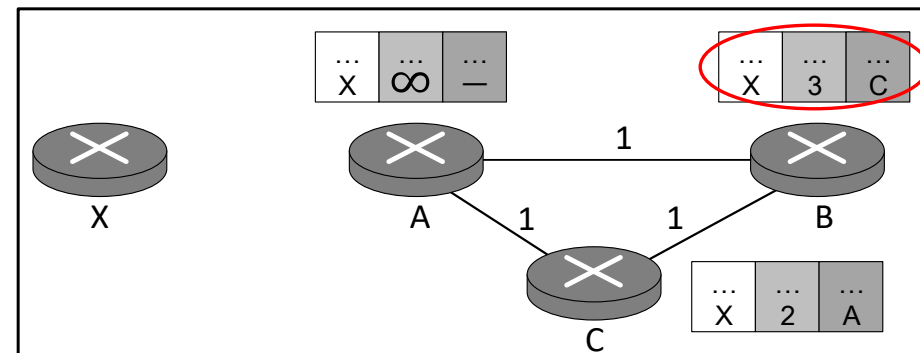
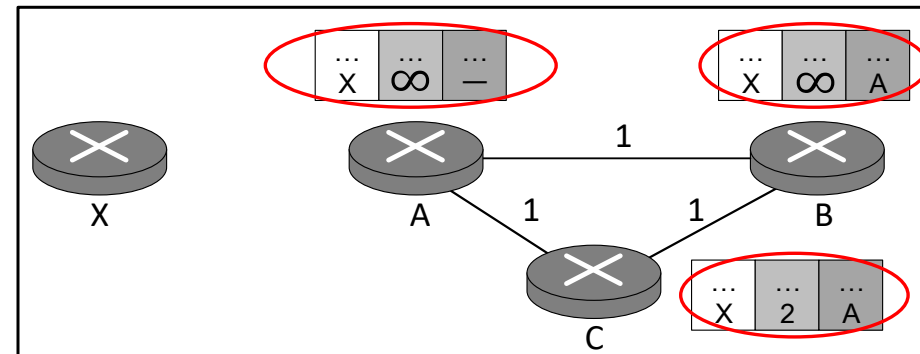
거리 벡터 라우팅

• 무한대로의 카운트(Count to Infinity) 문제 (5/6)

• 세 노드 루프 (1/2)

• 발생 과정

1. A가 X에 도달이 불가능한 것을 발견한 후, B, C에게 패킷을 전송해 상황을 알림
2. B는 테이블을 갱신했으나 C로 가는 패킷은 유실되어 C에 도달하지 못함
3. C는 B에게 라우팅 테이블을 보내 X로 가는 경로를 알리고 B는 테이블을 갱신함



거리 벡터 라우팅

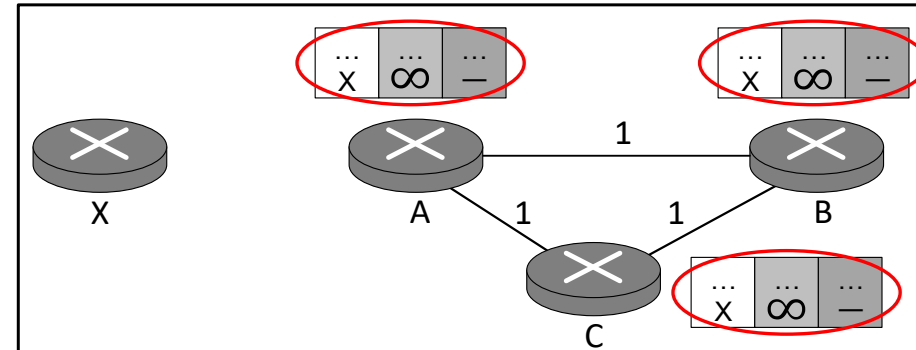
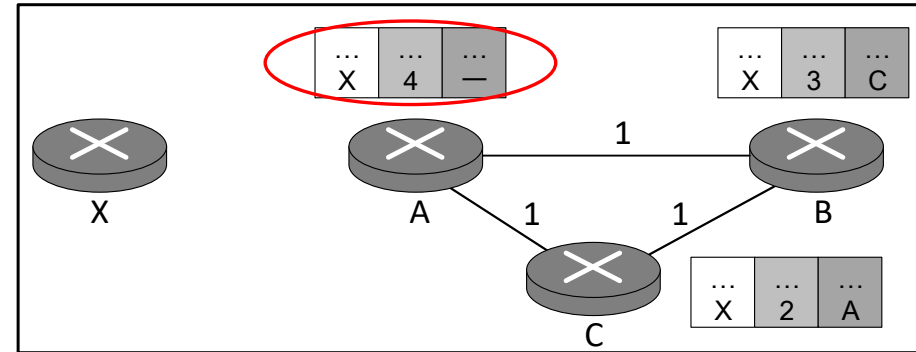
• 무한대로의 카운트(Count to Infinity) 문제 (6/6)

• 세 노드 루프 (2/2)

• 발생 과정

4. B는 라우팅 테이블을 A에게 전달, A는 테이블을 갱신함

5. 위 과정을 반복하고, 각 노드의 비용이 무한대가 되면 중지됨



• 해결 방안

• 두 노드 루프 해결 방법으로 해결이 보장되지 않아 새로운 해결 방법이 필요

• 라우트 포이즈닝(Route Poisoning)

• 특정 목적지 경로가 실패하거나 유효하지 않으면, 그 목적지에 대해 메트릭을 무한대로 설정해 모든 이웃에게 즉시 *광고하는 방식

*광고(Advertising)

라우터가 자신이 알고 있는 최신 네트워크 정보를 다른 라우터들에게 널리 알리는 것

거리 벡터 라우팅

- RIP (Routing Information Protocol) (1/11)

- 정의

- 홉 수를 기준으로 최단 경로를 결정하는 거리 벡터 방식의 동적 라우팅 프로토콜

- 특징

- 자율 시스템 내부에서 사용되는 도메인 내 라우팅 프로토콜
- 거리 벡터 라우팅에 기반을 두는 프로토콜
- 홉 수를 *메트릭으로 사용
- 무한대를 16으로 정의
 - 자율 시스템 내에서 15홉 이상의 경로를 가지지 못 함
- 다음 홉 열린 패킷이 목적지에 도달하기 위해 보내져야 하는 라우터의 주소를 정의

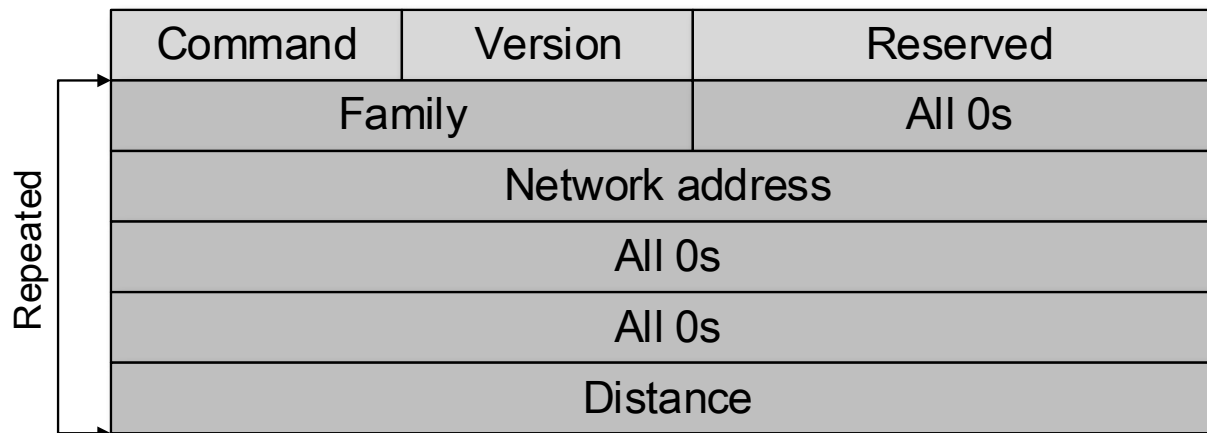
*메트릭(Metric)

라우터가 목적지 까지 가는 경로 중에서 어떤 경로가 가장 좋은 지 판단하는 기준 값

거리 벡터 라우팅

- RIP (Routing Information Protocol) (2/11)
- 메시지 형태

필드	크기	역할
명령(Command)	8비트	메시지의 종류를 표시 (1은 요청, 2는 응답)
버전(Version)	8비트	RIP 버전 표시
계열(Family)	16비트	사용된 프로토콜의 종류를 정의
네트워크 주소(Network address)	14비트	목적지 네트워크 주소 정의
거리(Distance)	32비트	송신 라우터로부터 목적지 네트워크까지의 홑 수를 표시



거리 벡터 라우팅

• RIP (Routing Information Protocol) (3/11)

• 요청

- 새로 생기거나 시간이 만료된 항목을 가진 라우터에 의해 전송
- 특정 경로나 모든 경로에 대해 요청 가능

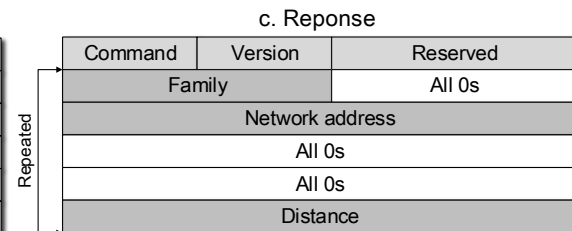
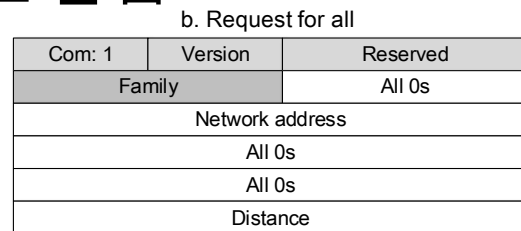
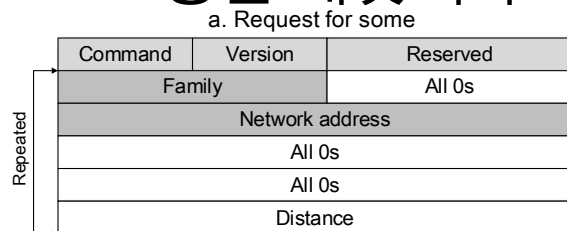
• 응답

• 요청된 응답

- 해당 요청에 대한 응답으로만 전송
- 해당 요청에서 지정된 목적지에 대한 정보를 포함

• 요청되지 않은 응답

- 모든 라우팅 테이블을 다루는 정보를 포함
- 30초 주기로 브로드캐스트
- 갱신 패킷이라고도 불림

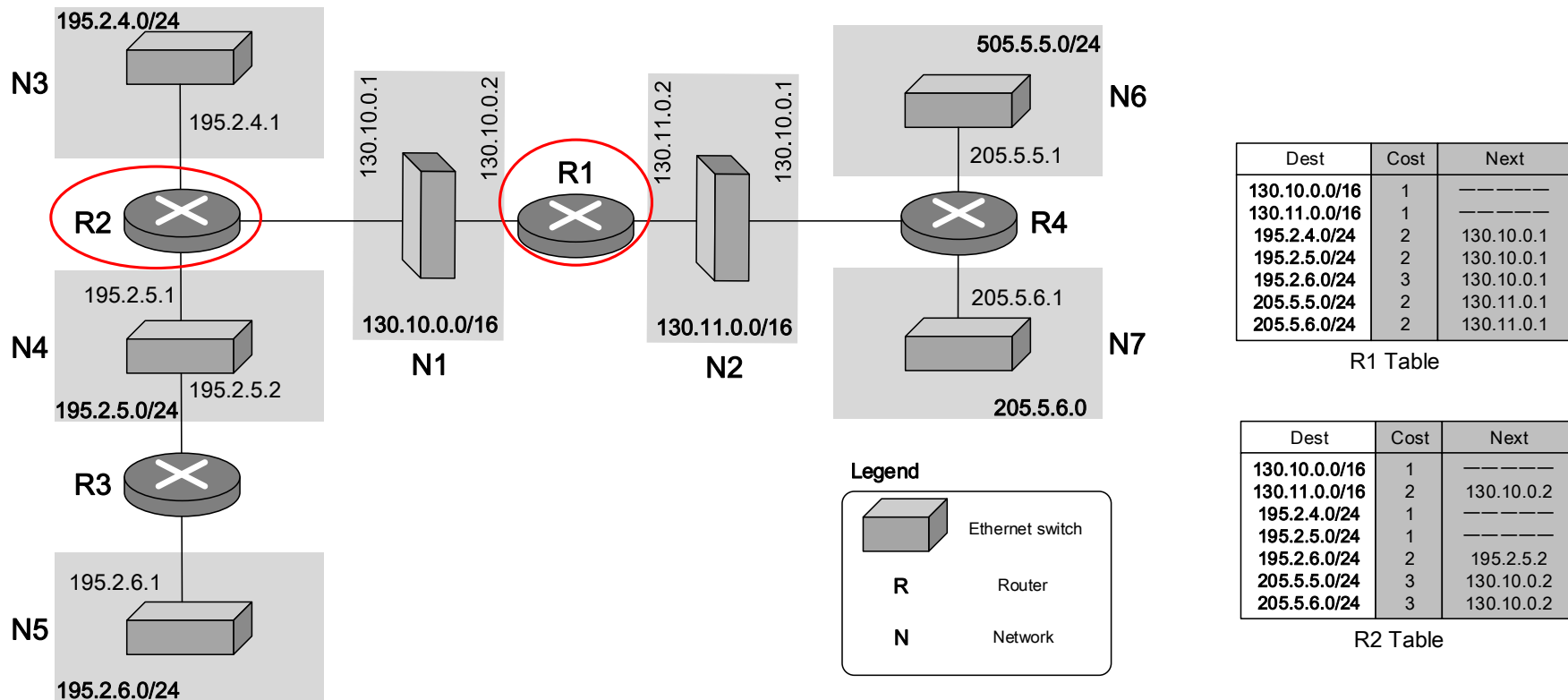


거리 벡터 라우팅

• RIP (Routing Information Protocol) (4/11)

• 예제 11.3 (1/2)

다음 상황에서 라우터 R1에서 R2로 보내지는 갱신 메시지를 보이고, 이에 대해 설명하여라.



거리 벡터 라우팅

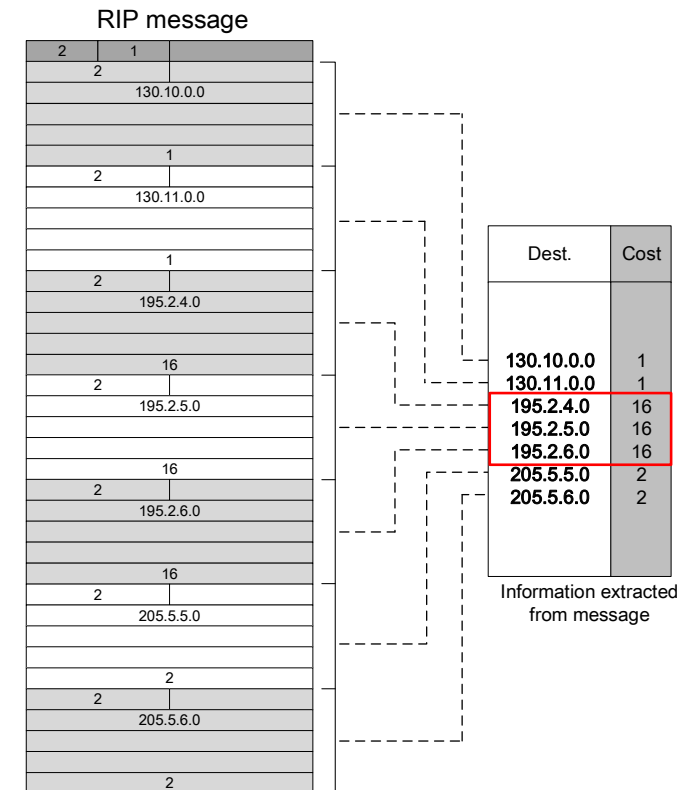
• RIP (Routing Information Protocol) (5/11)

• 예제 11.4 (2/2)

다음 상황에서 라우터 R1에서 R2로 보내지는 갱신 메시지를 보이고, 이에 대해 설명하여라.

• 풀이

- 수평 분할과 포이즌 리버스정책을 염두에 두고 R1은 195.2.4.0, 195.2.6.0 정보를 R2로 알렸음에 따라, 무한대 값인 16으로 대체함
- RIP 메시지를 전송하는 IP 데이터그램의 송신자 주소를 다음 홑 주소로 사용
- 응답 데이터는 R1을 기준으로 하는 값이므로, R2는 각 홑 수를 1씩 증가시켜서 계산함

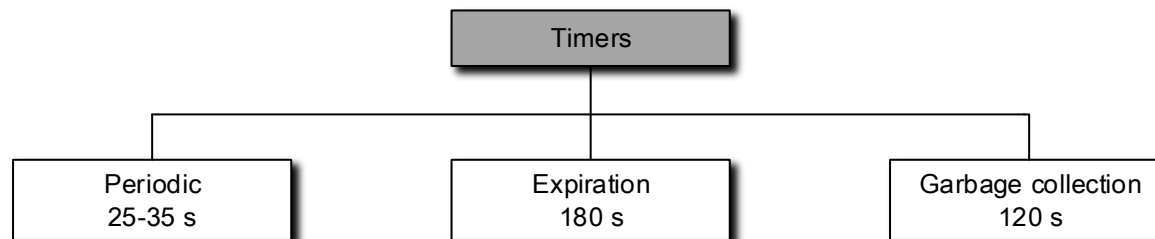


거리 벡터 라우팅

- RIP (Routing Information Protocol) (6/11)

- RIP 타이머 (1/2)

- RIP를 사용하는 라우터가 안정적으로 경로 정보를 관리하고 잘못된 정보로 인한 문제를 방지하기 위해 사용
 - e.g., 라우팅 루프, 무한대 계수 문제 등
- 종류
 - 주기적 타이머(Periodic Timer)
 - 정규 갱신 메시지의 통보 제어
 - 만료 타이머(Expiration Timer)
 - 경로의 유효성 관리
 - 폐 경로 수집 타이머(Garbage collection Timer)
 - 유효하지 않은 경로를 삭제하기 전에 통보



거리 벡터 라우팅

- RIP (Routing Information Protocol) (7/11)
 - RIP 타이머 (2/2)
 - 주기적 타이머(Periodic Timer)
 - 정규 갱신 메시지의 통보 제어
 - 25초~35초 사이의 임의의 값 사용
 - 라우터들이 동시에 갱신하고자 하는 경우에 과부하를 방지
 - 만료 타이머(Expiration Timer)
 - 라우터가 경로에 대한 갱신 메시지를 수신하게 되면 만료 타이머를 180초로 설정
 - 해당 경로에 대한 신규 갱신을 수신할 때마다 타이머 재설정
 - 180초 내에 도착하지 못하면 경로는 해제되어야 한다고 판단하고 홉 수를 16으로 설정
 - 폐 경로 수집 타이머(Garbage collection Timer)
 - 삭제 전에 이웃들에게 경로가 유효하지 않음을 알리기 위해서 사용
 - 해당 경로에 대해 120초로 설정
 - 해당 값이 0이 되면 표에서 삭제

거리 벡터 라우팅

• RIP (Routing Information Protocol) (8/11)

• RIPv2 (1/3)

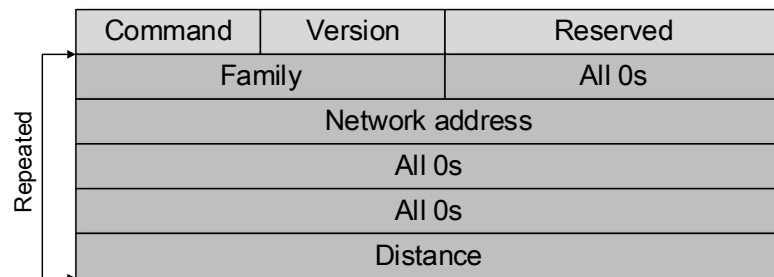
*CIDR(Classless Inter-Domain Routing)

라우터나 스위치가 들어온 데이터를 어디로 보내야 할 지 모를 때, 받은 포트를 제외한 모든 포트로 복사해서 보내는 전파 방식

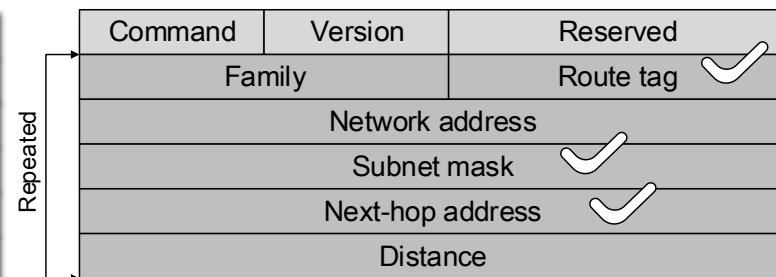
- 기존의 RIP 형식 중 TCP/IP 프로토콜에서 모두 0으로 채워졌던 필드들을 새로운 필드로 대체한 방식

필드	역할
경로 태그 (Route tag)	자율 시스템의 번호와 같은 정보를 전달 도메인 간 라우팅 프로토콜로부터 정보의 수신을 가능하게 함
서브넷 마스크 (Subnet Mask)	서브넷 마스크(또는 프리픽스)를 전달 클래스없는 주소와 *CIDR을 지원함을 의미
다음 홉 주소 (Next-hop address)	다음 홉의 주소를 알려줌 같은 네트워크를 공유할 경우에 유용

<RIPv1>



<RIPv2>



거리 벡터 라우팅

- RIP (Routing Information Protocol) (9/11)

- RIPv2 (2/3)

- 기존 RIP와의 차이점

- 클래스 없는 주소지정(Classless addressing)

- v1은 클래스 기반 주소지정을 사용하여 네트워크당 호스트 수가 정해져 있기 때문에 주소 낭비가 발생
 - v2에선 고정된 클래스를 없애고, IP 주소를 필요한 만큼 할당하도록 설정

- 인증(Authentication)

- 가짜 라우터가 보내는 정보에 대해 메시지를 보호하기 위해 추가됨
 - 항목에 경로 정보가 아니라 인증 정보임을 알리기 위해 계열 필드를 1로 채움
 - 두 번째 필드인 인증 유형은 인증에 사용되는 방법을 나타냄
 - 세 번째 필드는 실제 인증 데이터를 포함함

Command	Version	Reserved
0xFFFF		Authentication
Authentication data 16 bytes		
⋮		

거리 벡터 라우팅

- RIP (Routing Information Protocol) (10/11)
 - RIPv2 (3/3)
 - 기존 RIP와의 차이점
 - 멀티캐스팅(Multicasting)
 - v1에서는 모든 이웃에게 RIP 메시지를 전달하기 위해 브로드캐스팅을 사용
 - v2에서는 네트워크에 있는 라우터들만 RIP 메시지를 수신할 수 있도록 멀티캐스트 주소 사용

거리 벡터 라우팅

- RIP (Routing Information Protocol) (11/11)
- 캡슐화(Encapsulation)
 - RIPv2 메시지는 UDP 사용자 데이터그램에 캡슐화됨
 - RIPv2 메시지는 길이를 나타내는 필드를 가지고 있지 않음
 - UDP 헤더에는 전체 길이를 알려주는 Length 필드가 존재함
 - RIPv2에 할당된 잘 알려진 포트는 UDP 520번 포트임



목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

링크 상태 라우팅

- 정의

- 각 라우터가 네트워크의 모든 연결 정보를 받아 지도를 제작, 지도를 바탕으로 최단 경로를 직접 계산하는 방식

- 특징

- 다익스트라(Dijkstra)알고리즘 사용
- RIP보다 빠른 변화 감지 및 대응
 - 네트워크에 장애가 발생하면 변화된 정보만 즉시 라우터에게 전파
- OSPF (Open Shortest Path Test) 프로토콜로 구현

링크 상태 라우팅

- 라우팅 테이블 생성 (1/7)
 - 각 노드가 다른 노드로 가는 최소 비용 노드를 보여주는 라우팅 테이블을 가질 수 있도록 4단계를 거침
 1. 링크 상태 패킷(LSP, Link State Packet) 생성
 - 각 노드에 의해 링크 상태를 생성하는 것
 2. LSP 플러딩(Flooding)
 - 모든 다른 라우터로 효과적이며 안전한 방법으로 LSP들을 발송하는 것
 3. 다익스트라(Dijkstra) 알고리즘 기반의 최단 경로 설정
 4. 라우팅 테이블 계산

링크 상태 라우팅

• 라우팅 테이블 생성 (2/7)

*토폴로지(Topology)
네트워크의 구조 또는 지도

1. 링크 상태 패킷(LSP)이 생성되는 경우

- 도메인의 *토폴로지에 변화가 있는 경우
 - 도메인 내의 임의의 노드에 토폴로지를 갱신하도록 함
- 주기적으로 LSP를 생성하는 경우
 - 오래된 정보가 도메인에서 제거되도록 보장함

링크 상태 라우팅

- 라우팅 테이블 생성 (3/7)

2. LSP 플러딩

- 노드가 LSP를 준비한 후 모든 다른 노드들에게 발송하는 절차
- 동작 과정
 1. LSP를 생성한 노드는 각 인터페이스를 통해 LSP의 복사본을 전송
 2. LSP를 수신한 노드는 이전에 받았던 복사본과 비교한 후 새로 받은 것이 이전 것보다 더 오래된 것이라면 해당 LSP는 폐기
 3. 새로 받은 것이 이전 것보다 더 새 것이라면 다음 과정을 수행
 1. 오래된 LSP를 버리고 새 것을 가짐
 2. 라우터는 인터페이스에서 수신한 LSP를 라우터가 가진 나머지 포트들을 통해 전송하여 플러딩이 도메인 내에서 중지되는 것을 보장

링크 상태 라우팅

- 라우팅 테이블 생성 (4/7)

- 3. 다익스트라(Dijkstra) 알고리즘 기반의 최단 경로 설정

- 최단 경로 트리를 생성하는 알고리즘

- 동작 과정

- 1. 초기화

- 지정된 노드를 트리의 루트로 설정하고 이를 경로에 추가
 - 루트의 모든 이웃에 대한 최단 경로를 루트와 그 이웃들 간의 비용으로 설정
 - 루트로의 최단 거리는 0으로 설정

- 2. 반복 수행

- 경로 상에 없는 노드들을 검색한 후 최단 거리를 가지는 것을 선택하여 경로에 추가
 - 2번째 절차에서 경로에 추가된 노드의 최단 거리를 사용하여 모든 나머지 노드들의 최단 거리를 갱신

링크 상태 라우팅

• 라우팅 테이블 생성 (5/7)

3. 다익스트라(Dijkstra) 알고리즘 기반의 최단 경로 설정

• 알고리즘 코드

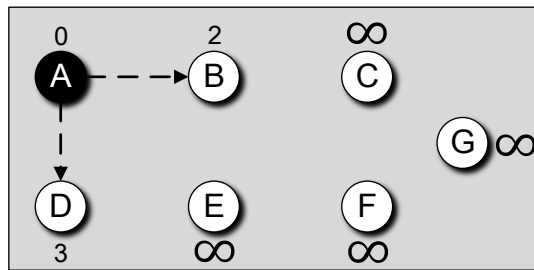
- 초기화 과정에서 A는 자신을 루트로 설정
- 토폴로지 상의 각 노드로의 최단 경로 거리 할당
- A의 이웃이 아닌 노드들은 무한대의 값이 할당
- 각 반복 과정에서 최소 거리를 가지는 다음 노드가 선택되고 경로 상에 추가
- $Weight(u,v)$ 는 u에서 v로 가는 길의 비용을 표시

```
1. Dijkstra(Graph, source):  
  
2.     // 초기화  
3.     for each vertex v in Graph:  
4.         distance[v] ← ∞  
5.         visited[v] ← false  
6.         distance[source] ← 0  
  
7.     // 반복  
8.     while (모든 노드가 확정되지 않을 때까지):  
  
9.         u ← 방문하지 않은 노드 중 distance 값이 가장  
작은 노드  
10.        visited[u] ← true  
  
11.        for each neighbor v of u:  
12.            if not visited[v]:  
13.                // 갱신 조건  
14.                if distance[v] > distance[u] +  
weight(u, v): // 출발점에서 v까지의 최단 거리와  
출발점에서 u까지의 최단 거리 + u에서 v까지 가는 거리의  
합 비교  
15.                    distance[v] ← distance[u] +  
weight(u, v)  
16.                // ← 여기서 "갱신 발생"  
  
17.        // (Iteration 종료 시점마다)  
18.        // 확정된 노드 집합과 현재 distance 테이블을  
기록
```

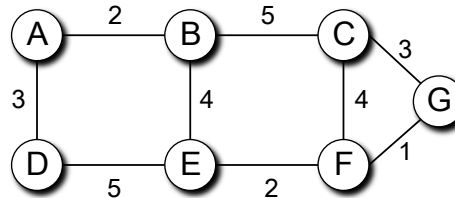
링크 상태 라우팅

• 라우팅 테이블 생성 (6/7)

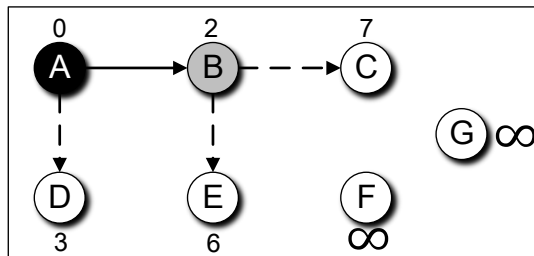
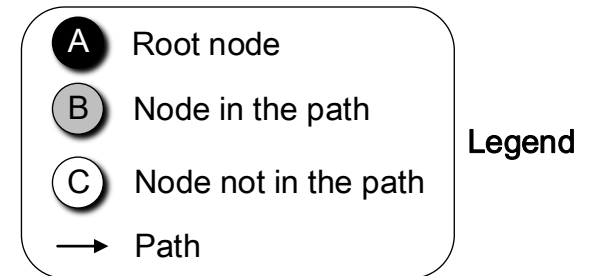
3. 다익스트라(Dijkstra) 알고리즘 기반의 최단 경로 설정



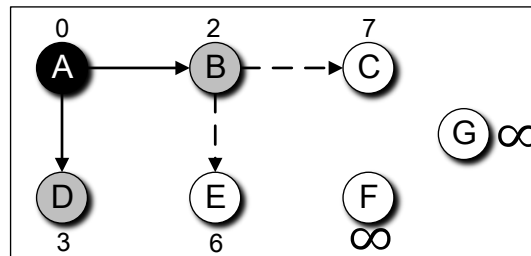
Initialization



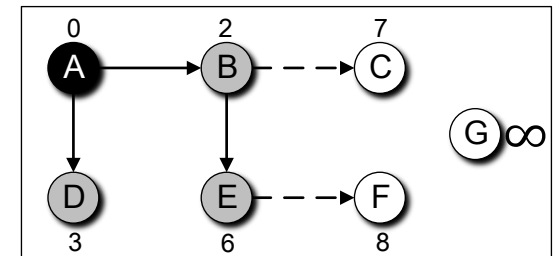
Topology



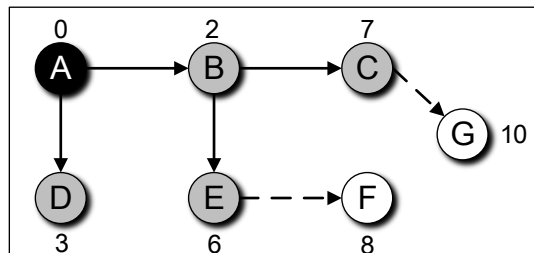
Iteration 1



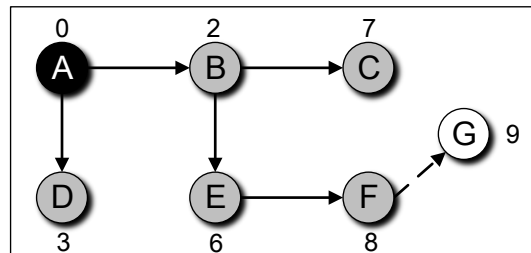
Iteration 2



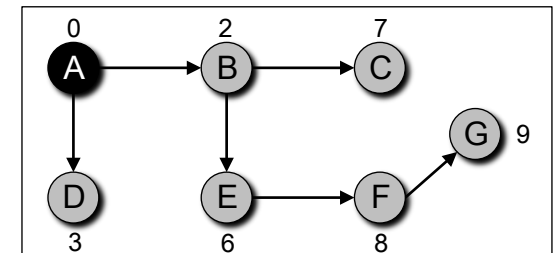
Iteration 3



Iteration 4



Iteration 5



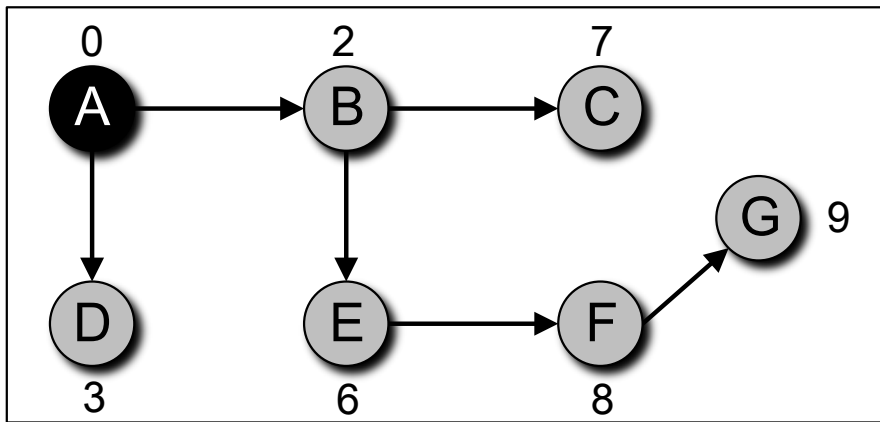
Iteration 6

링크 상태 라우팅

• 라우팅 테이블 생성 (7/7)

4. 라우팅 테이블 계산

- 최단 경로 트리로부터 라우팅 테이블의 계산
 - 각 노드는 다익스트라 알고리즘을 기반으로 한 최단 경로 트리 프로토콜을 사용하여 자신의 라우팅 테이블구성



Destination	Cost	Next Router
A	0	-
B	2	-
C	7	B
D	3	-
E	6	B
F	8	B
G	9	B

링크 상태 라우팅

- OSPF (Open Shortest Path First) (1/35)

- 정의

- 링크 상태 라우팅에 근거를 둔 도메인 내 라우팅 프로토콜

*지역(areas)

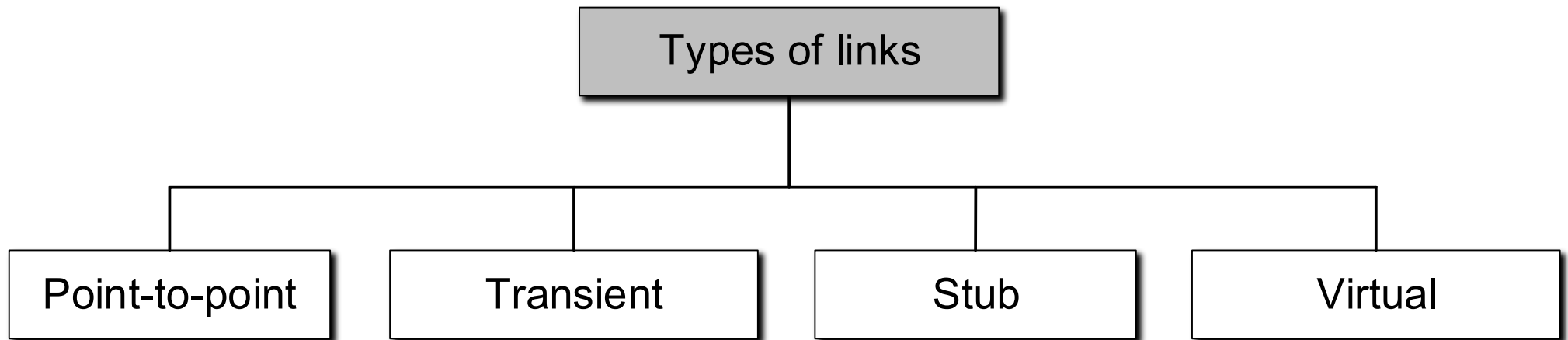
자율 시스템 내의 포함되는 호스트,
라우터 및 네트워크 모음

- 특징

- 자율 시스템을 여러 *지역(areas)으로 분할
 - 지역 내 라우터는 경로 정보 지역 내로 플러딩

링크 상태 라우팅

- OSPF (Open Shortest Path First) (2/35)
 - 링크의 유형 (1/6)
 - 점-대-점 링크(Point-to-point Link)
 - 경유 링크(Transient Link)
 - 스템브 링크(Stub Link)
 - 가상 링크(Virtual Link)



링크 상태 라우팅

- OSPF (Open Shortest Path First) (3/35)

- 링크의 유형 (2/6)

- 점-대-점 링크(Point-to-point Link)

- 정의

- 라우터 사이에 어떤 다른 호스트나 라우터 없이 직접 연결하는 링크

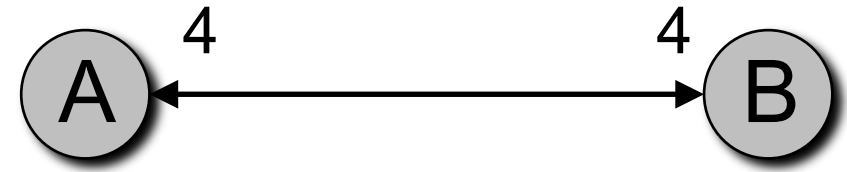
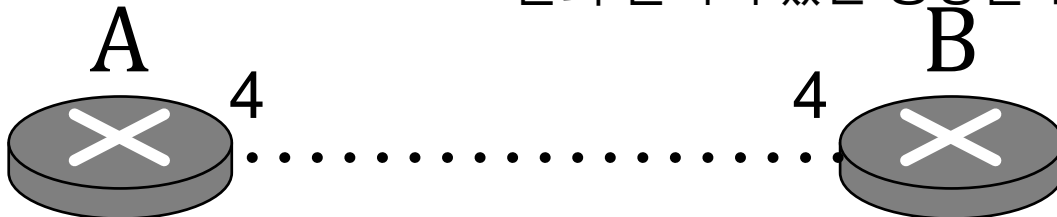
- 특징

- 여러 장치가 회선을 나눠 쓰지 않아 대역폭이 보장됨
 - 목적지가 하나여서 MAC 주소를 찾기 위한 과정이 생략됨
 - 라우터는 노드로 표현되고 링크는 노드 간을 연결하는 양방향 화살표로 표현

- 활용 예시

- 기업의 본사-지사 간 전용선

- 멀리 떨어져 있는 공장을 연결해야 할 때, 점-대-점 링크를 구축해 사용



링크 상태 라우팅

• OSPF (Open Shortest Path First) (4/35)

• 링크의 유형 (3/6)

• 경유 링크(Transient Link)

• 정의

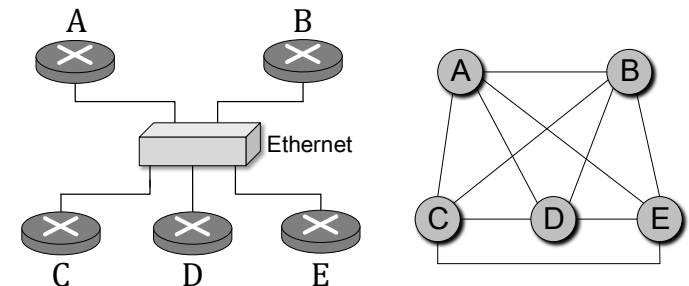
- 두 대 이상의 라우터가 함께 연결된 공유 네트워크를 나타내는 링크

• 특징

- 데이터는 어떤 라우터로도 들어갈 수 있고 어떤 라우터로부터 나갈 수 있음
- 모든 LAN과 2개 이상의 라우터를 가지는 일부 WAN이 해당됨
- 공유 네트워크에 연결된 모든 라우터가 일대일로 통신하면 라우터 수가 늘어날수록 통신량이 기하급수적으로 증가하여 비효율적임
 - 5개의 라우터가 있으면 한 번에 업데이트에 20번의 통신이 필요함
- 네트워크의 한 라우터가 지정(Designated) 라우터 역할을 하여 효율성을 높임

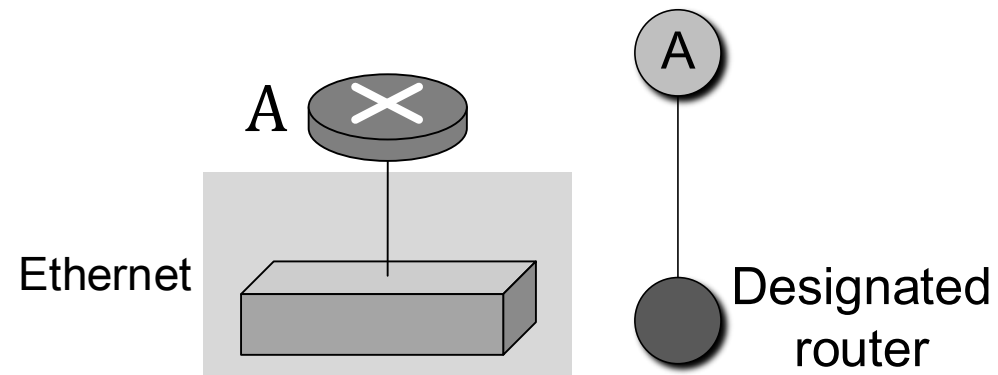
• 활용 예시

- Ethernet LAN 같은 여러 개의 라우터가 LAN 장비를 통해 같은 네트워크에 접속



링크 상태 라우팅

- OSPF (Open Shortest Path First) (5/35)
 - 링크의 유형 (4/6)
 - 스템브 링크(Stub Link)
 - 정의
 - 오직 하나의 라우터에만 연결된 네트워크
 - 특징
 - 데이터 패킷은 단일 라우터를 통해서만 네트워크와 통신 가능
 - 링크가 라우터에서 네트워크로의 단 방향만을 가짐
 - 독립적인 네트워크를 구성하여 안정성과 자율성을 확보할 수 있음
 - 활용 예시
 - 사이버-물리 시스템(CPS, Cyber-Physical System)



링크 상태 라우팅

- OSPF (Open Shortest Path First) (6/35)

- 링크의 유형 (5/6)

- 스템브 링크(Stub Link)

- 사이버-물리 시스템(CPS, Cyber-Physical System)

- 물리적 시스템과 컴퓨터자원 같은 사이버 시스템을 통합하여 상호작용하는 시스템

- 구성 요소

- 물리적 프로세스

- 기계, 공장 설비, 발전소, 차량 등

- 센서

- 데이터를 수집하고 동작을 제어

- 네트워크

- 실시간 데이터 전송

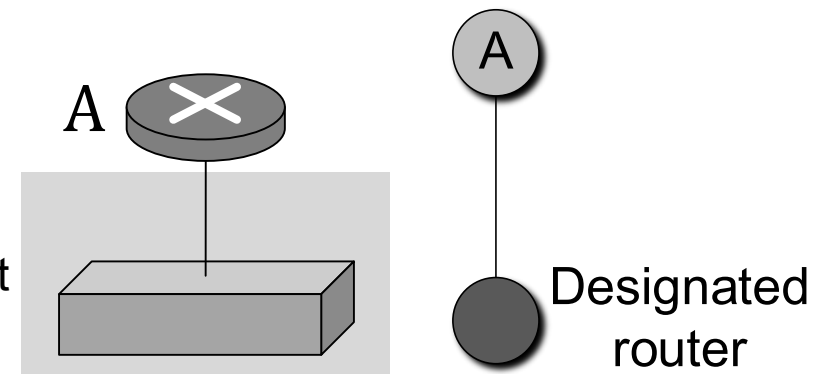
- 사이버 시스템

- 제어 로직, 데이터 분석 등

- 운영자

- 모니터링 및 의사결정

Ethernet



링크 상태 라우팅

- OSPF (Open Shortest Path First) (7/35)

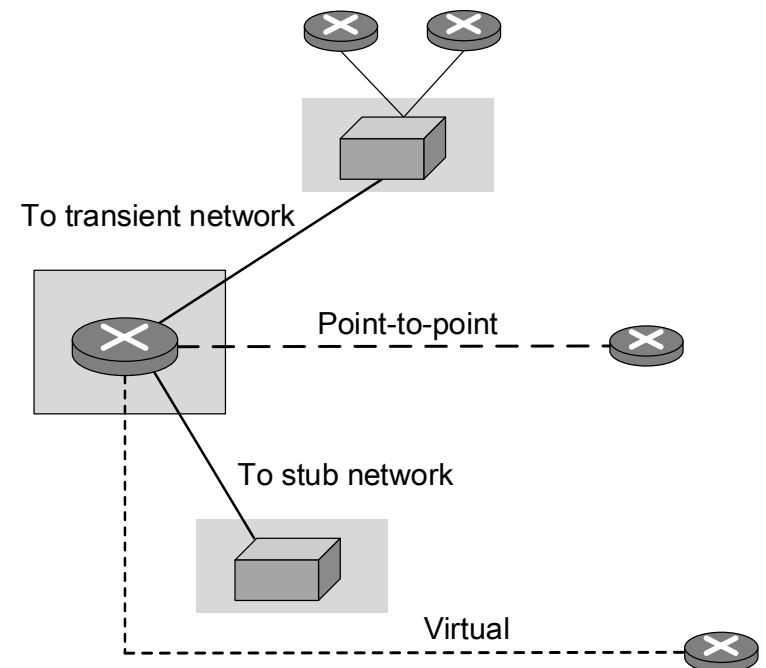
- 링크의 유형 (6/6)

- 가상 링크(Virtual Link)

- 정의

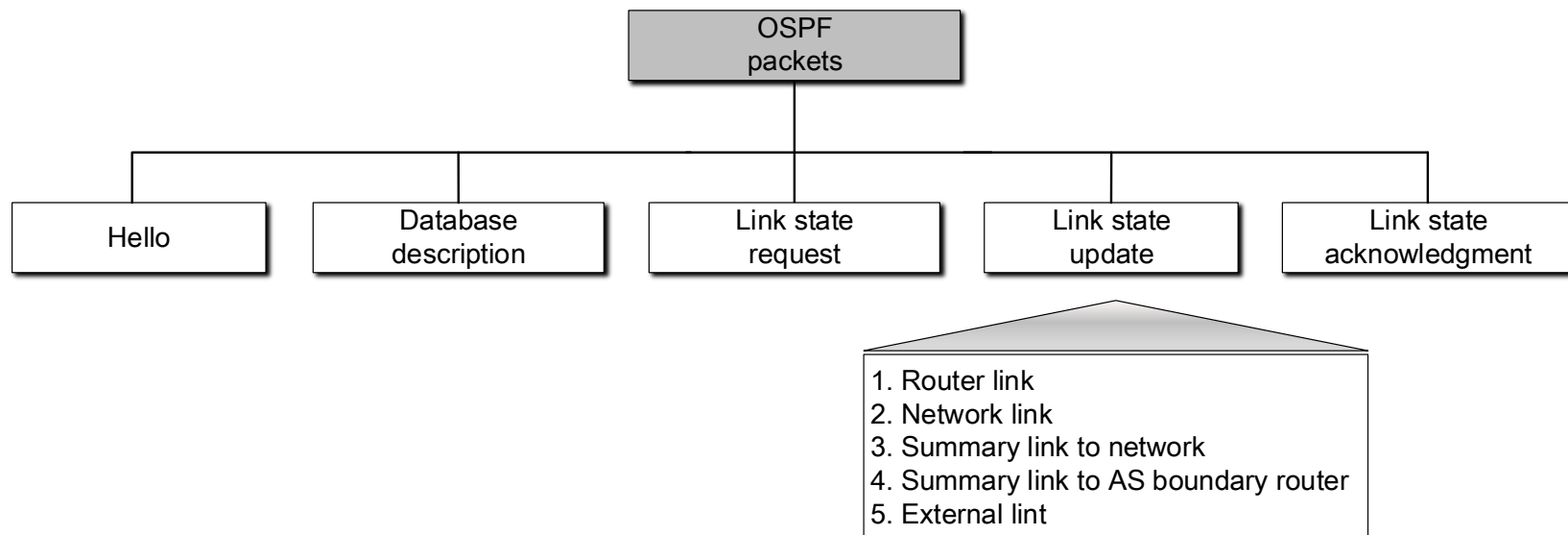
- 두 라우터 간의 연결이 끊어졌을 때를 대비하여 작동하는 여러 라우터들 사이에 연결된 링크

- 네 가지 링크 유형 모식도



링크 상태 라우팅

- OSPF (Open Shortest Path First) (8/35)
- OSPF 패킷 유형 (1/11)
 - Hello 패킷
 - 데이터베이스 기술(Description) 패킷
 - 링크 상태 요청 패킷
 - 링크 상태 갱신 패킷
 - 링크 상태 확인응답 패킷



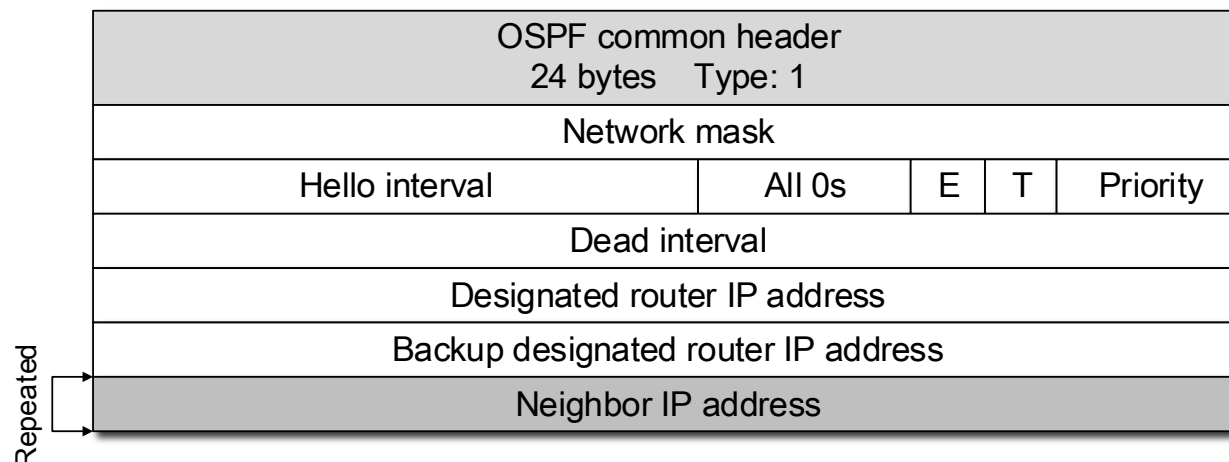
링크 상태 라우팅

- OSPF (Open Shortest Path First) (9/35)
- OSPF 패킷 유형 (2/11)
 - Hello 패킷
 - 정의
 - 이웃 관계를 생성하고 이웃의 도달 가능성을 검사하는 패킷
 - 특징
 - 링크 상태 라우팅의 첫 번째 과정
 - 라우터가 자신의 이웃에 대한 정보를 다른 라우터로 전송하기 전에 먼저 자신의 이웃과 인사를 해야만 함

링크 상태 라우팅

- OSPF (Open Shortest Path First) (10/35)
- OSPF 패킷 유형 (3/11)
 - Hello 패킷

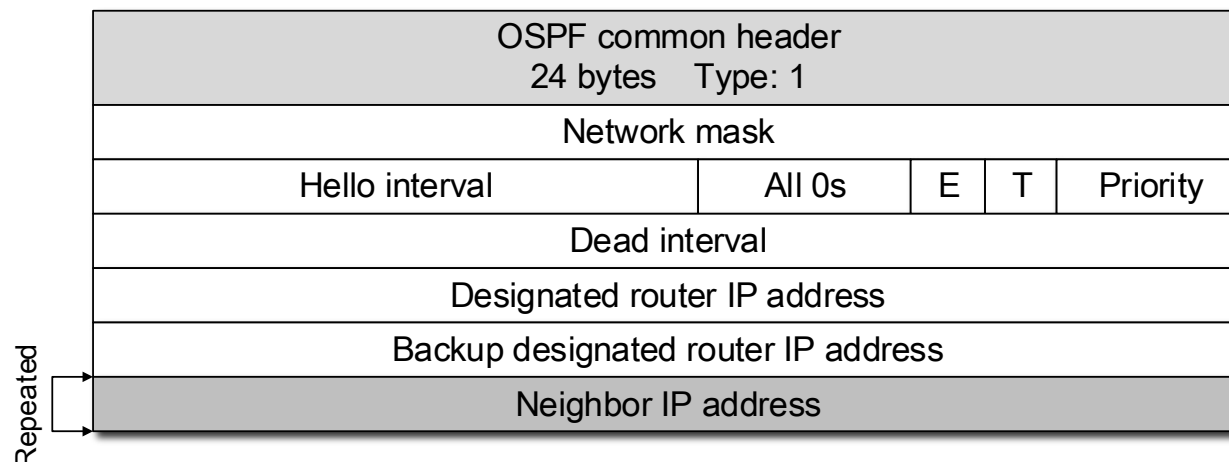
필드	크기	역할
네트워크 마스크 (Network mask)	32 비트	hello 메시지가 전송되어지는 네트워크의 네트워크 마스크를 표시
Hello 간격(Interval)	16 비트	hello 메시지 간의 초 단위 설정
E 플래그(E flag)	1 비트	1로 설정되어있으면 지역이 스타브 지역이 됨
F 플래그(F flag)	1 비트	1로 설정되면 라우터는 다중 메트릭을 지원
우선 순위(Priority)	8 비트	라우터의 우선순위를 표시 우선순위는 지정 라우터와 백업 지정 라우터를 결정



링크 상태 라우팅

- OSPF (Open Shortest Path First) (11/35)
- OSPF 패킷 유형 (4/11)
 - Hello 패킷

필드	크기	역할
Dead 간격(Dead Interval)	32	이웃이 다운되었다고 판단하기 위해, Hello 메시지를 받지 못한 시간이 해당 간격을 넘으면 시간을 초 단위로 나타냄
지정 라우터 IP 주소 (Designed router IP address)	32	메시지가 전송되는 네트워크에 대한 지정 라우터의 IP 주소를 나타냄
백업 지정 라우터 IP 주소 (Backup designated IP address)	32	메시지가 전송되는 네트워크에 대한 백업 지정 라우터의 IP 주소를 나타냄
이웃 IP 주소 (Neighbor IP address)	32	Dead 간격 내에 hello 패킷을 송신한 모든 이웃 라우터들의 IP 주소를 나타냄



링크 상태 라우팅

- OSPF (Open Shortest Path First) (12/35)
- OSPF 패킷 유형 (5/11)
 - 데이터베이스 기술(Description) 패킷
 - 정의
 - *데이터베이스의 전체 윤곽에 대한 정보만을 보내는 패킷
 - 특징
 - 새로 연결된 라우터는 어떤 라인이 정보를 안 가지고 있는지 탐색
 - 두 라우터가 데이터베이스 기술 메시지를 교환하기 원하면 하나가 주(Master)가 되고 다른 하나는 종(Slave)이 됨

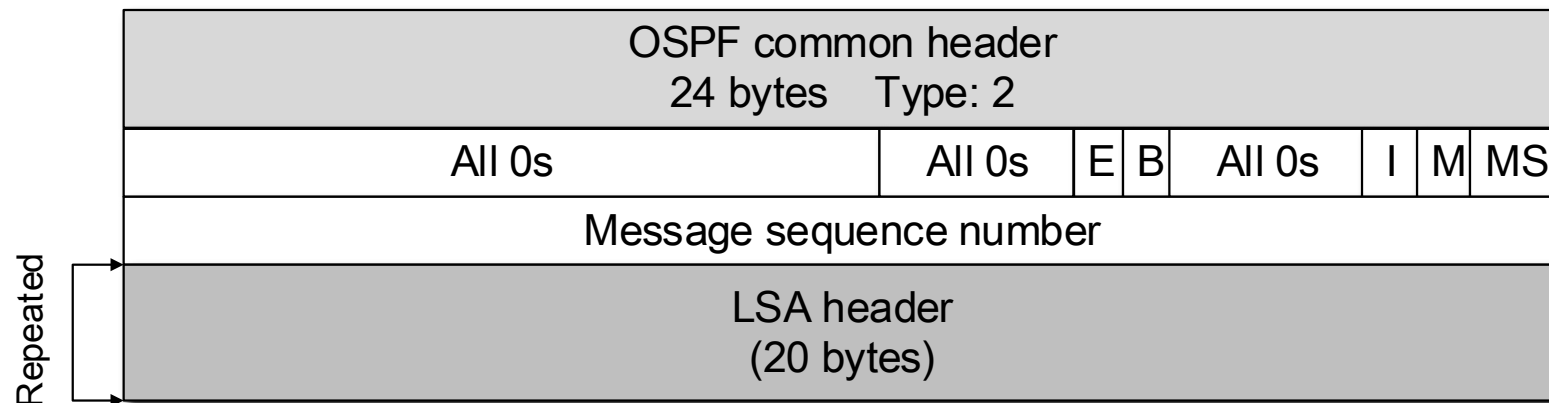
*데이터베이스(Database)

주어진 목적이나 주어진 자료 처리 시스템에 사용하기에 적합하도록 자료를 구조화하여 자료 검색 및 갱신을 효율화한 자료의 집합, 여기서 OSPF가 내부적으로 관리하는 라우팅 관련 데이터 구조

링크 상태 라우팅

- OSPF (Open Shortest Path First) (13/35)
- OSPF 패킷 유형 (6/11)
 - 데이터베이스 기술(Description) 패킷

필드	크기	역할
E 플래그(External flag)	1 비트	광고 라우터가 자율 시스템 경계 라우터이면 1로 지정됨
B 플래그(B flag)	1 비트	광고 라우터가 지역 경계 라우터일 경우 1로 지정됨
I 플래그(Initialization flag)	1 비트	메시지가 첫 메시지일 때 1로 지정
M 플래그(More flag)	1 비트	마지막 메시지가 아닐 경우에 1로 지정
M/S 플래그 (Master/Slave)	1 비트	패킷의 시작점을 나타냄 1로 지정되면 주를, 0으로 지정되면 종임을 표시

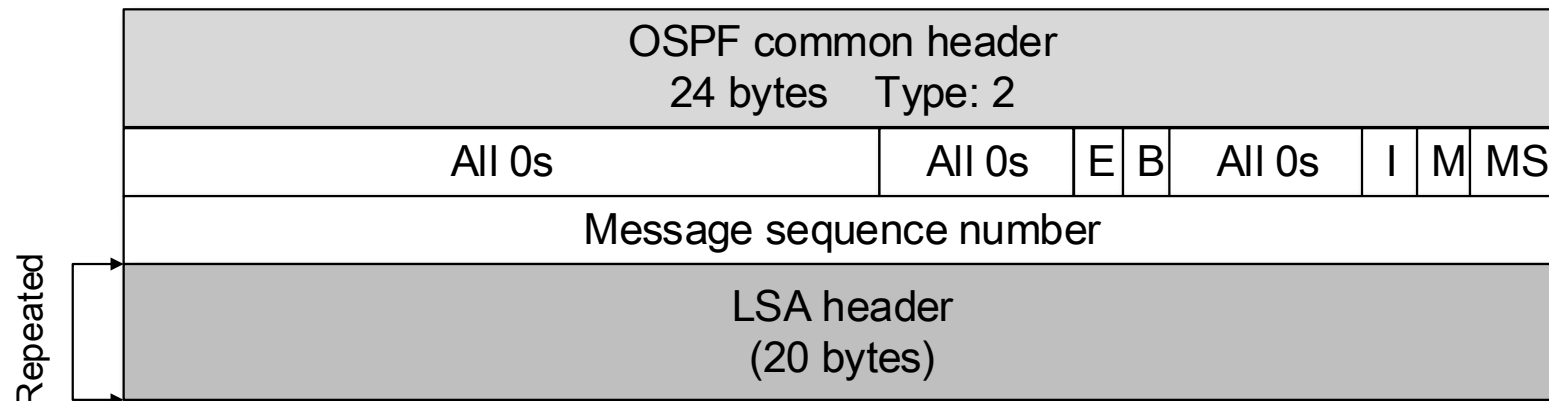


링크 상태 라우팅

- OSPF (Open Shortest Path First) (14/35)
- OSPF 패킷 유형 (7/11)
 - 데이터베이스 기술(Description) 패킷

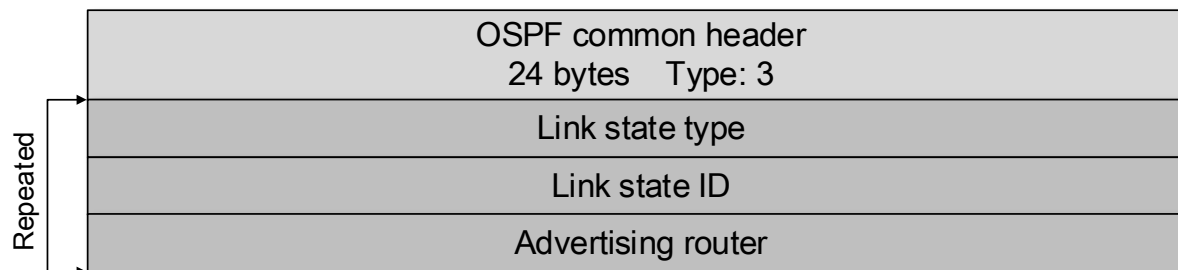
필드	크기	역할
메시지 순서 번호	32 비트	메시지의 순서 번호를 포함하여 요청에 대한 응답 비교
*LSA 헤더	20 바이트	각 LSA에 사용 각 링크의 운곽을 알려주며, 링크 상태 데이터베이스의 각 링크에 대해 반복

*링크 상태 광고(LSA, Link State Advertisement)
라우터가 자신이 알고 있는 링크 상태(연결 관계, 네트워크 정보 등)을 이웃 라우터에게 알릴 때 사용하는 메시지



링크 상태 라우팅

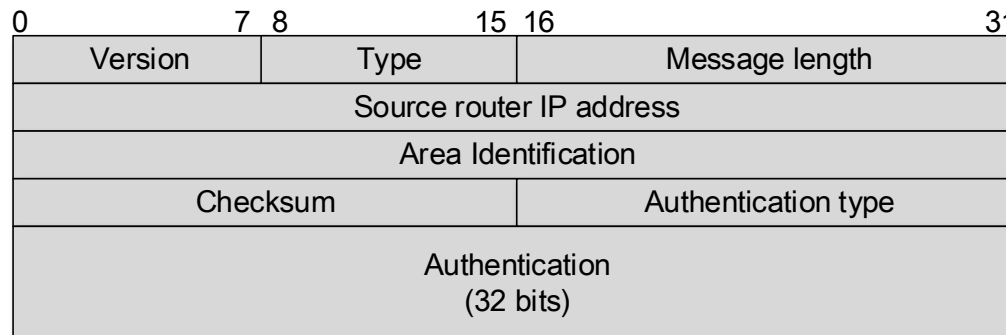
- OSPF (Open Shortest Path First) (15/35)
- OSPF 패킷 유형 (8/11)
 - 링크 상태 요청 패킷(Link state request packet)
 - 정의
 - 특정 경로나 경로들에 대한 정보를 필요로 하는 라우터에 의해 보내어 지는 패킷
 - 특징
 - 링크 상태 갱신 패킷으로 응답
 - 새로 연결된 라우터가 데이터베이스 기술 패킷을 수신한 후 몇 개의 경로들에 대한 더 많은 정보를 요청할 때 사용
 - 형식
 - 세 개의 필드로 이루어진 각 세트는 하나의 단일 LSA에 대한 요청
 - 세트는 하나 이상의 광고가 필요한 경우 반복



링크 상태 라우팅

- OSPF (Open Shortest Path First) (16/35)
- OSPF 패킷 유형 (9/11)
 - 공통 헤더 (1/2)
 - OSPF 패킷은 공통 헤더를 공유

필드	크기	역할
버전(Version)	8 비트	프로토콜의 버전 정의
유형(Type)	8 비트	패킷의 유형 표시(1~5)
메시지 길이(Message length)	16 비트	헤더를 포함한 전체 메시지 길이 표시
발신지 라우터 IP 주소 (Source router IP address)	32 비트	패킷을 송신한 라우터의 IP주소 표시
지역 식별자 (Area identification)	32 비트	라우팅이 일어나는 지역 표시



링크 상태 라우팅

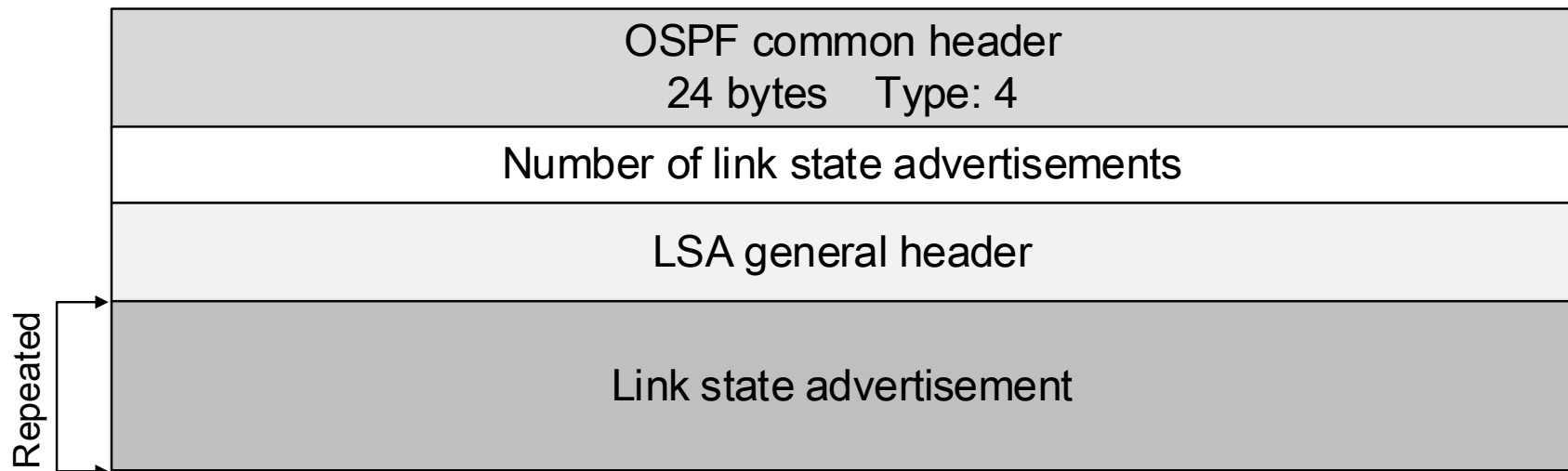
- OSPF (Open Shortest Path First) (17/35)
- OSPF 패킷 유형 (10/11)
 - 공통 헤더 (1/2)
 - OSPF 패킷은 공통 헤더를 공유

필드	크기	역할
체크섬	16 비트	인증 종류와 인증 데이터를 제외한 전체 패킷의 오류 검사에 사용
인증 종류(Authentication type)	16 비트	지역에서 사용되는 인증 방법 정의 (0은 아무것도 사용하지 않음 1은 패스워드를 사용함)
인증 (Authentication)	64 비트	인증 데이터의 실제 값 (인증 종류가 0이면 0으로 채워짐, 인증 종류가 1이면 8문자의 패스워드가 포함됨)

0	7	8	15	16	31
Version		Type		Message length	
Source router IP address					
Area Identification					
Checksum			Authentication type		
Authentication (32 bits)					

링크 상태 라우팅

- OSPF (Open Shortest Path First) (18/35)
- OSPF 패킷 유형 (11/11)
 - 링크 상태 갱신 패킷(Link state update packet)
 - 라우터가 링크의 상태를 광고하는데 사용
 - 특징
 - 각 갱신 패킷이 서로 다른 LSA를 포함할 수 있음
 - 모든 5개의 LSA 패킷들은 공통의 일반적인 헤더를 가짐



링크 상태 라우팅

- OSPF (Open Shortest Path First) (19/35)

- 링크 상태 갱신 패킷 (1/3)
 - 공통 헤더

필드	역할
링크 상태 시간 (Link state age)	메시지가 처음 만들어지고 난 후부터 경과된 시간을 초 단위로 나타냄 초기 값은 0 연속해서 다른 라우터가 이 메시지를 이어서 전송하게 되면 경유 시간을 추정 후 필드의 값에 더함
E 플래그(E flag)	스터브 필드일 경우 1로 지정
T 플래그(T flag)	라우터가 여러 개의 서비스 유형을 처리할 수 있으면 1로 지정

Link state age	Reserved	E	T	Link state type
Link state ID				
Advertising router				
Link state sequence number				
Link state checksum	Length			

링크 상태 라우팅

- OSPF (Open Shortest Path First) (20/35)

- 링크 상태 갱신 패킷 (2/3)

- 공통 헤더

필드	역할
링크 상태 유형 (Link state type)	5가지 LSA 유형을 표시 라우터 링크(1), 네트워크 링크(2), 네트워크 요약 링크(3), 자율 시스템 경계 라우터 요약 링크(4), 외부 링크(5)
링크 상태 식별자 (Link state ID)	링크의 유형에 의존하여 결정 (1) – 라우터의 IP 주소, (2) – 지정 라우터의 IP 주소, (3) – 네트워크의 주소 (4) – 자율 시스템 경계 라우터의 IP 주소, (5) – 외부 네트워크 주소

Link state age	Reserved	E	T	Link state type
Link state ID				
Advertising router				
Link state sequence number				
Link state checksum	Length			

링크 상태 라우팅

• OSPF (Open Shortest Path First) (21/35)

- 링크 상태 갱신 패킷 (3/3)
 - 공통 헤더

*광고(Advertising)

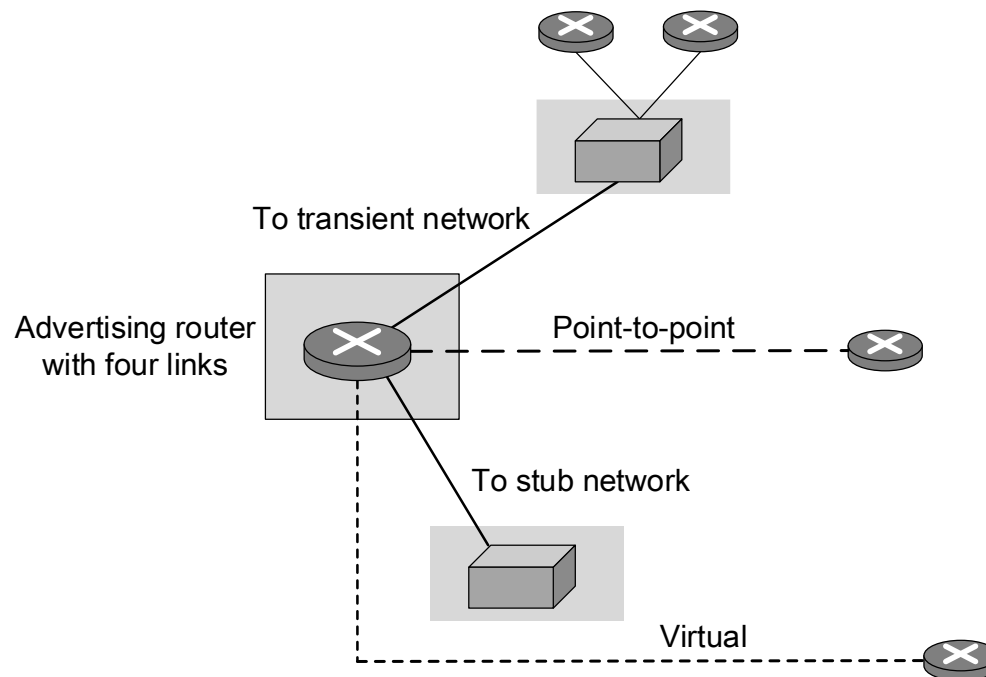
라우터가 자신이 알고 있는 최신 네트워크 정보를 다른 라우터들에게 널리 알리는 것

필드	역할
광고 라우터 (Advertising router)	메시지를 *광고하는 라우터의 IP 주소
링크 상태 순서 번호 (Link state sequence number)	각 링크 상태 갱신 메시지에 할당된 순서 번호
링크 상태 검사합 (Link state checksum)	경과시간 필드를 제외한 전체 패킷에 *폴레처의 검사합이라는 방법 사용
길이(Length)	전체 패킷의 길이를 바이트 단위로 표시

Link state age	Reserved	E	T	Link state type
Link state ID				
Advertising router				
Link state sequence number				
Link state checksum	Length			

링크 상태 라우팅

- OSPF (Open Shortest Path First) (22/35)
 - 라우터 링크 LSA (1/4)
 - 정의
 - 실제 라우터의 링크들을 정의하는 패킷
 - 특징
 - 라우터는 이 LSA를 사용하여 라우터의 모든 링크들에 관해 광고함

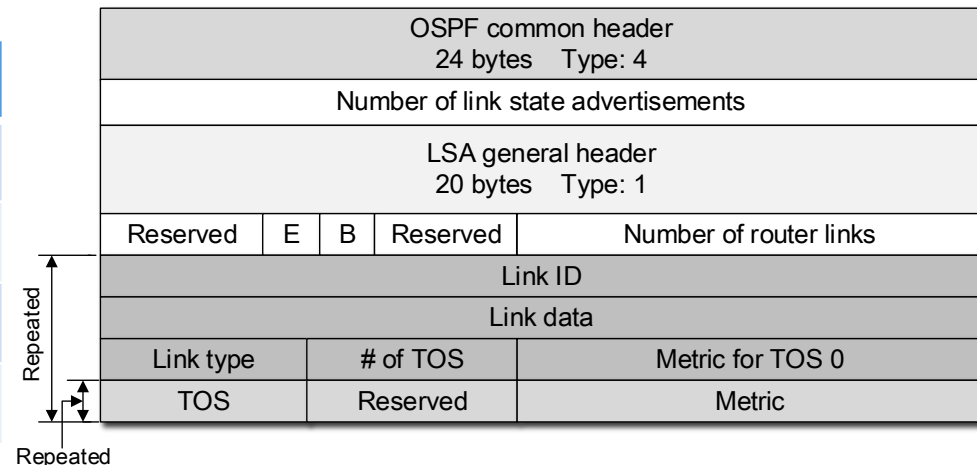


링크 상태 라우팅

- OSPF (Open Shortest Path First) (23/35)
 - 라우터 링크 LSA (2/4)
 - 패킷 형식

필드	역할
링크 식별자(Link ID)	링크의 유형에 따라 다름
링크 데이터(Link Data)	링크에 관한 추가 정보 제공 링크의 유형에 따라 다름
링크 유형(Link type)	라우터가 연결된 네트워크 유형에 따라 4가지의 서로 다른 유형의 링크 유형 정의

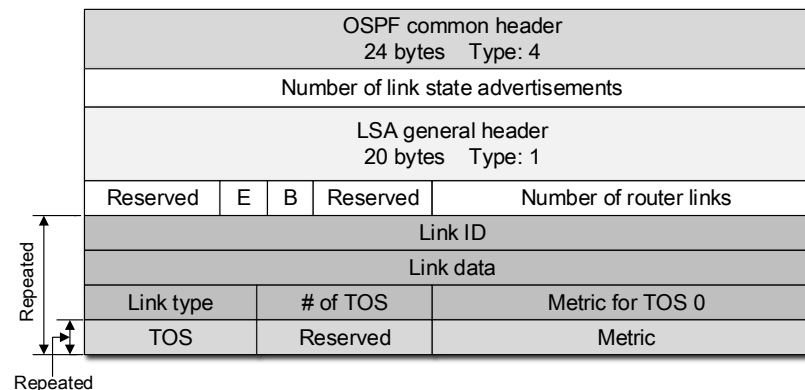
링크 유형	링크 식별자	링크 데이터
Type 1: 점대점	이웃 라우터의 주소	인터페이스 번호
Type 2: 경유	지정 라우터의 주소	라우터 주소
Type 3: 스텔브	네트워크 주소	네트워크 마스크
Type 4: 가상	이웃 라우터의 주소	라우터 주소



링크 상태 라우팅

- OSPF (Open Shortest Path First) (24/35)
- 라우터 링크 LSA (3/4)
 - 패킷 형식

필드	역할
서비스 유형 (TOS, Type Of Service)	각 링크에 알려진 서비스 유형의 개수 표시
TOS 0에 대한 메트릭 (Metric for TOS 0)	기본 서비스인 TOS 0에 대한 메트릭 표시
TOS	서비스 유형 표시
메트릭(Metric)	해당 TOS에 대한 메트릭 표시



링크 상태 라우팅

• OSPF (Open Shortest Path First) (25/35)

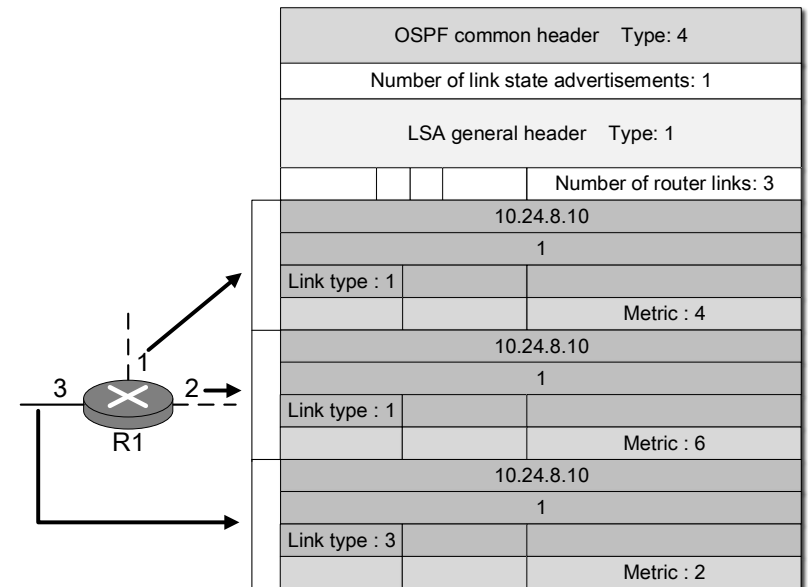
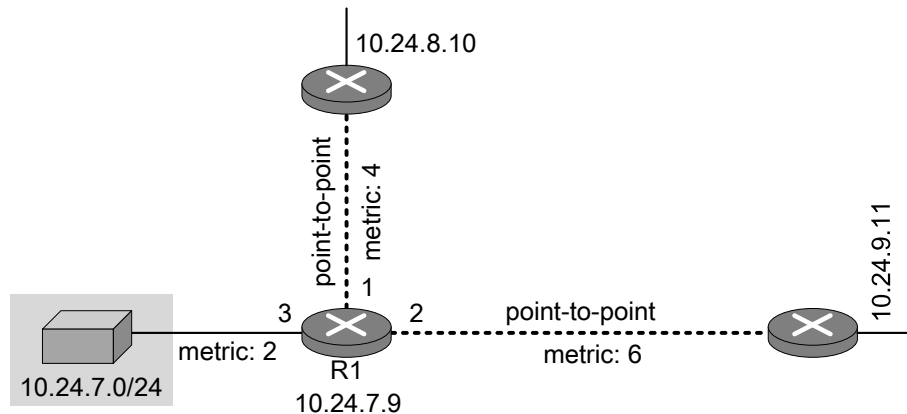
• 라우터 링크 LSA (4/4)

• 예제 11.4

라우터 10.24.7.9에 의해 전송되는 라우터 링크 LSA를 나타내어라.

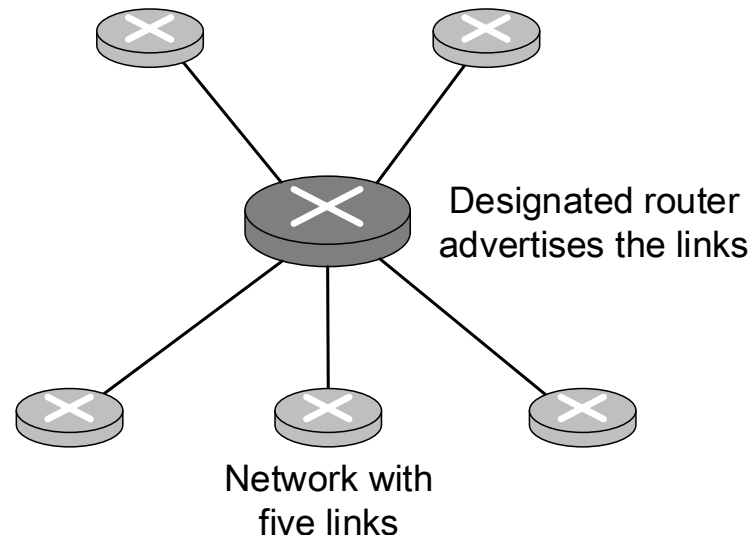
• 풀이

- 라우터는 3개의 링크를 가짐
 - 하나는 type 3(stub)이고,
둘은 type 1(point-to-point)임
- 다음 그림과 같이 표현될 수 있음



링크 상태 라우팅

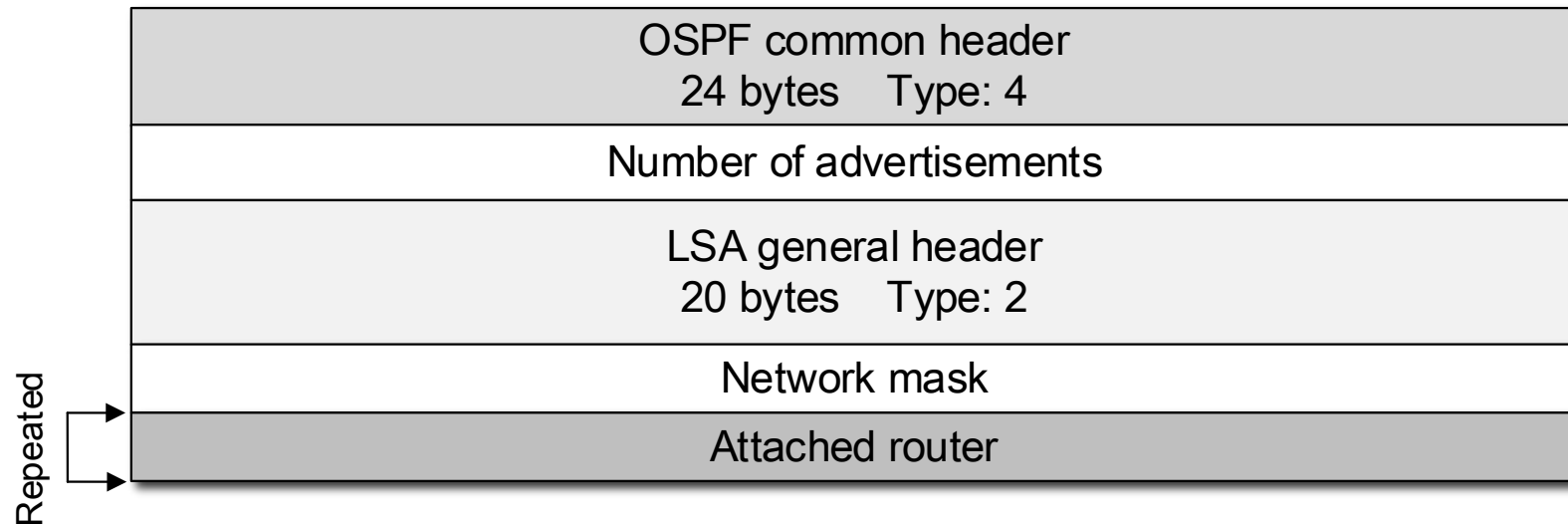
- OSPF (Open Shortest Path First) (26/35)
 - 네트워크 링크 LSA (1/4)
 - 정의
 - 네트워크 링크들을 정의하는 링크
 - 특징
 - 지정 라우터는 경유 망(Transient network)을 대신 해 이런 유형의 LSP 패킷들을 분배함
 - 네트워크에 연결된 모든 라우터들의 존재를 알림



링크 상태 라우팅

- OSPF (Open Shortest Path First) (27/35)
- 네트워크 링크 LSA (2/4)
 - 패킷 형식

필드	역할
네트워크 마스크(Network mask)	네트워크 마스크를 나타냄
접속 라우터(Attached router)	반복되는 필드로서 네트워크에 접속된 모든 라우터의 IP 주소들을 나타냄



링크 상태 라우팅

- OSPF (Open Shortest Path First) (28/35)

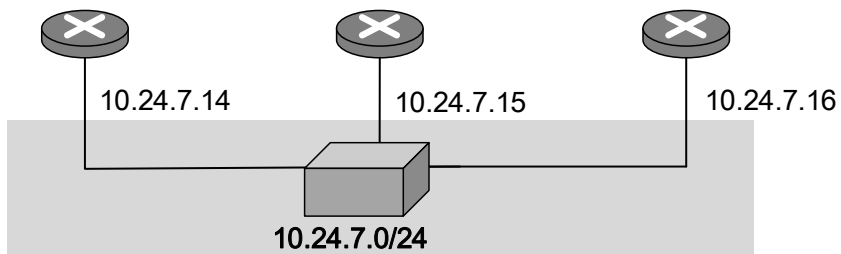
- 네트워크 링크 LSA (3/4)

- 예제 11.5

다음 그림에서 네트워크 링크 LSA를 나타내어라.

- 풀이

- 네트워크 링크를 광고하는 네트워크에는 라우터가 3개 접속되어 있음
 - 오른쪽 그림에서 LSA는 마스크와 라우터 주소를 보여줌



OSPF common header Type: 4
Number of advertisements: 1
LSA general header Type: 2
255.255.255.0
10.24.7.14
10.24.7.15
10.24.7.16

링크 상태 라우팅

- OSPF (Open Shortest Path First) (29/35)

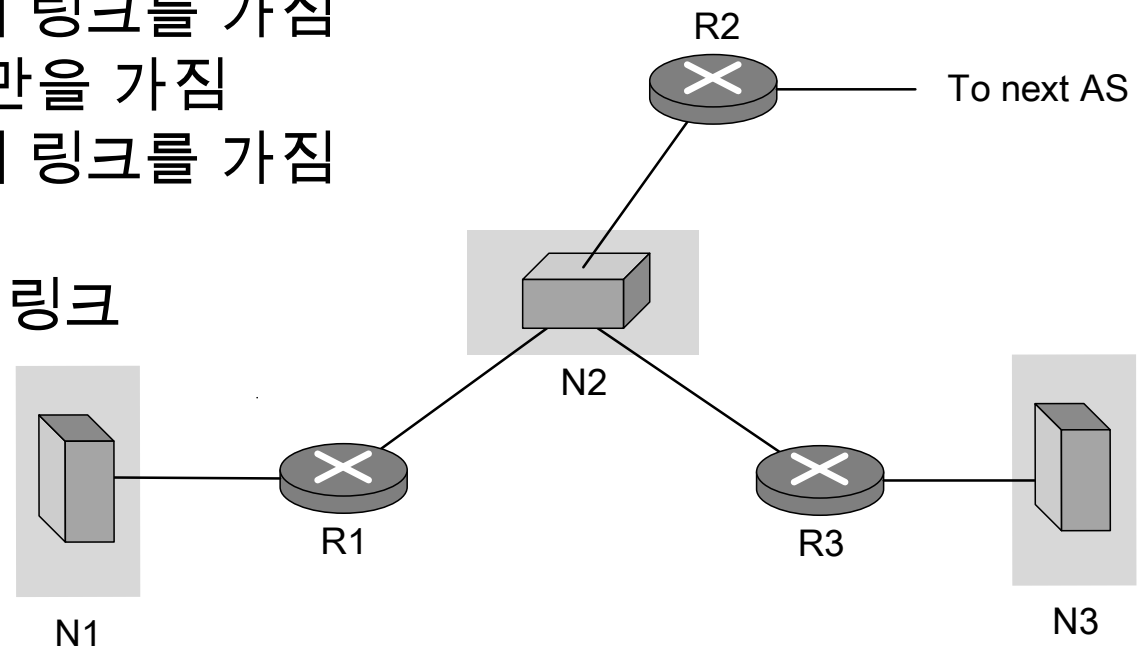
- 네트워크 링크 LSA (4/4)
 - 예제 11.6

다음 그림에서 어느 라우터가 LSA를 전송하는가?

- 풀이

- R1은 N1과 N2 두 개의 링크를 가짐
- R2는 N2 하나의 링크만을 가짐
- R3는 N2와 N3 두 개의 링크를 가짐

∴ 모든 라우터가 라우터 링크
LSA들을 광고함



링크 상태 라우팅

- OSPF (Open Shortest Path First) (30/35)

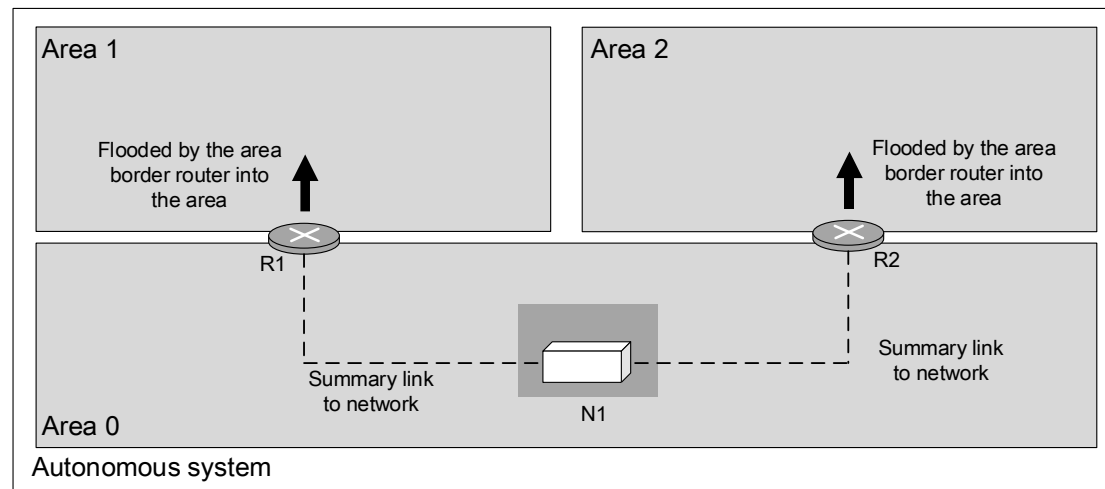
- 네트워크로의 요약 링크 LSA (1/2)

- 정의

- 지역 바깥의 다른 망 존재를 알리기 위해 지역 경계 라우터에 의해 사용되는 LSA

- 특징

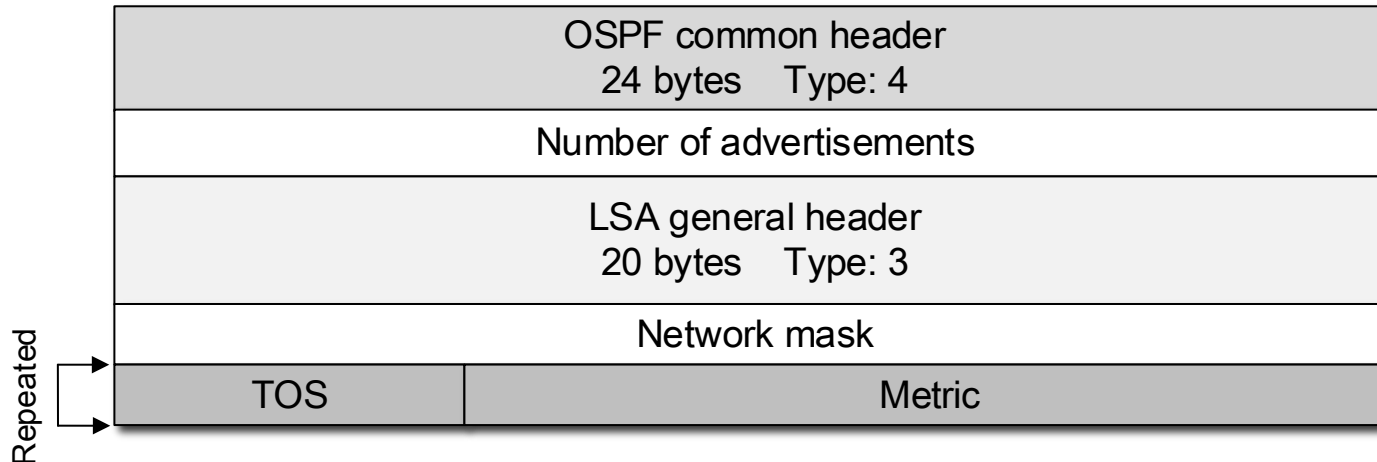
- 네트워크 마스크와 각 서비스 종류에 대한 메트릭으로 구성됨
 - 각 광고는 하나의 단일 네트워크만을 알림
 - 광고하는 라우터의 IP 주소는 링크 상태 광고의 헤더에 포함되어 알려짐



링크 상태 라우팅

- OSPF (Open Shortest Path First) (31/35)
- 네트워크로의 요약 링크 LSA (2/2)
 - 패킷 형식

필드	역할
네트워크 마스크(Network mask)	네트워크 마스크를 표시
TOS	서비스 유형을 표시
메트릭 (Metric)	TOS 필드에서 정의된 서비스 유형에 대한 메트릭을 표시



링크 상태 라우팅

- OSPF (Open Shortest Path First) (32/35)

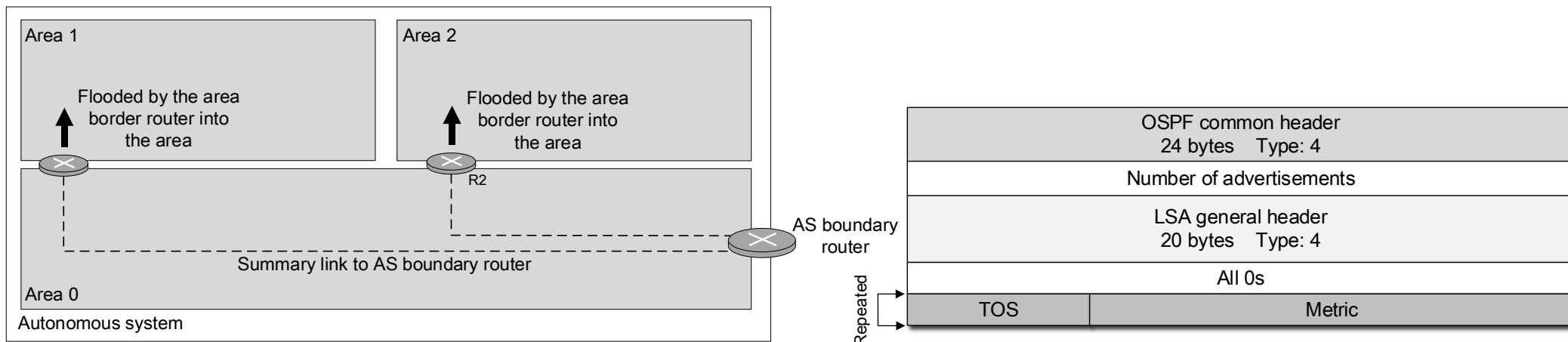
- 자율 시스템 경계 라우터로의 요약 링크 LSA (1/2)

- 정의

- 자율 시스템 경계 라우터로의 경로를 제공하는 LSA

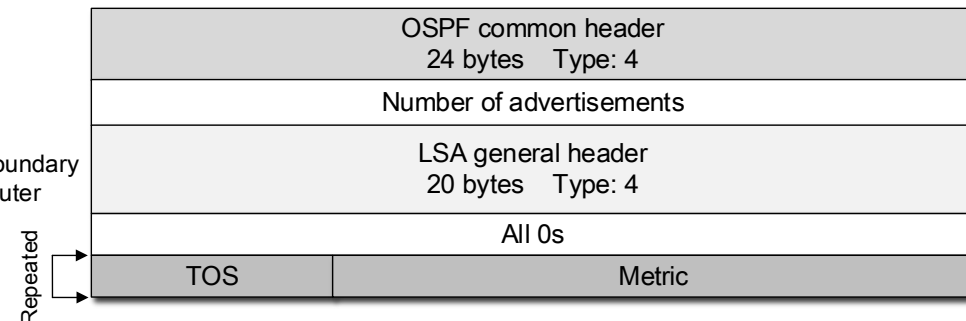
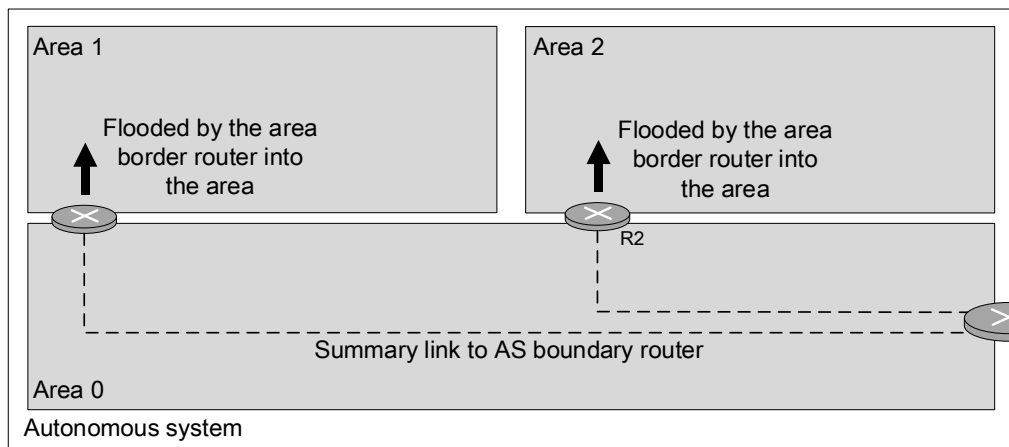
- 특징

- 지역 내의 라우터가 자율 시스템 바깥으로 패킷을 전송하고자 할 때 필요
 - 자율 시스템 외부로 전송될 메시지가 이 네트워크에 도달하면 자율 경계 라우터는 메시지를 수신



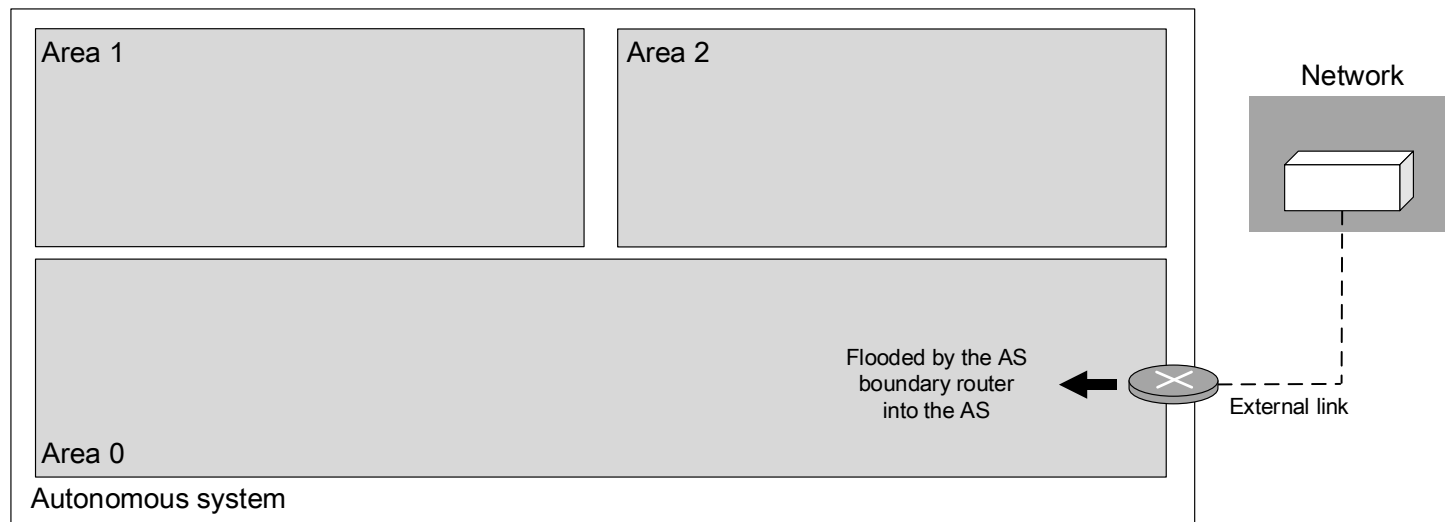
링크 상태 라우팅

- OSPF (Open Shortest Path First) (33/35)
 - 자율 시스템 경계 라우터로의 요약 링크 LSA (2/2)
 - 사용자가 다른 Area로 이동하는 경우
 - 같은 서브넷에 속한 Area로 이동하는 경우
 - Area 1과 Area 2는 직접 연결되어 있지 않지만, Area 0를 통해 간접적으로 연결되어 있어 Area 간 통신 경로가 존재하여 통신을 유지함
 - 다른 서브넷에 속한 Area로 이동하는 경우
 - 새로운 IP를 할당 받아야 함
 - 기존 세션이 유효하지 않아 일시적으로 통신이 중단됨



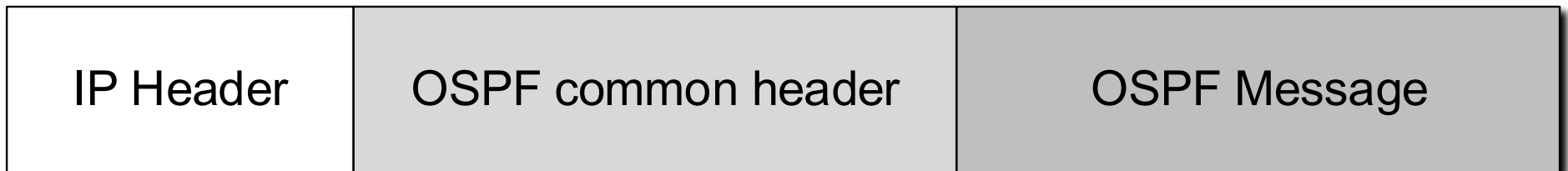
링크 상태 라우팅

- OSPF (Open Shortest Path First) (34/35)
 - 외부 링크 LSA
 - 정의
 - 자율 시스템 외부에 있는 모든 네트워크를 알리는 LSA
 - 특징
 - 자율 시스템 바깥에 이용 가능한 네트워크가 무엇인지 궁금할 때 사용
 - 각 광고는 하나의 네트워크만을 알림



링크 상태 라우팅

- OSPF (Open Shortest Path First) (35/35)
 - 캡슐화
 - OSPF는 IP 데이터그램에 캡슐화 됨
 - OSPF는 흐름 제어와 오류 제어에 관한 확인 응답 절차를 포함하기 때문에 이에 관한 서비스를 제공하는 전송 계층 프로토콜을 필요로 하지 않음



목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

경로 벡터 라우팅

- 정의

- 목적지에 도달하기 위해 거쳐가야 하는 전체 경로의 목록을 공유하는 방식

- 특징

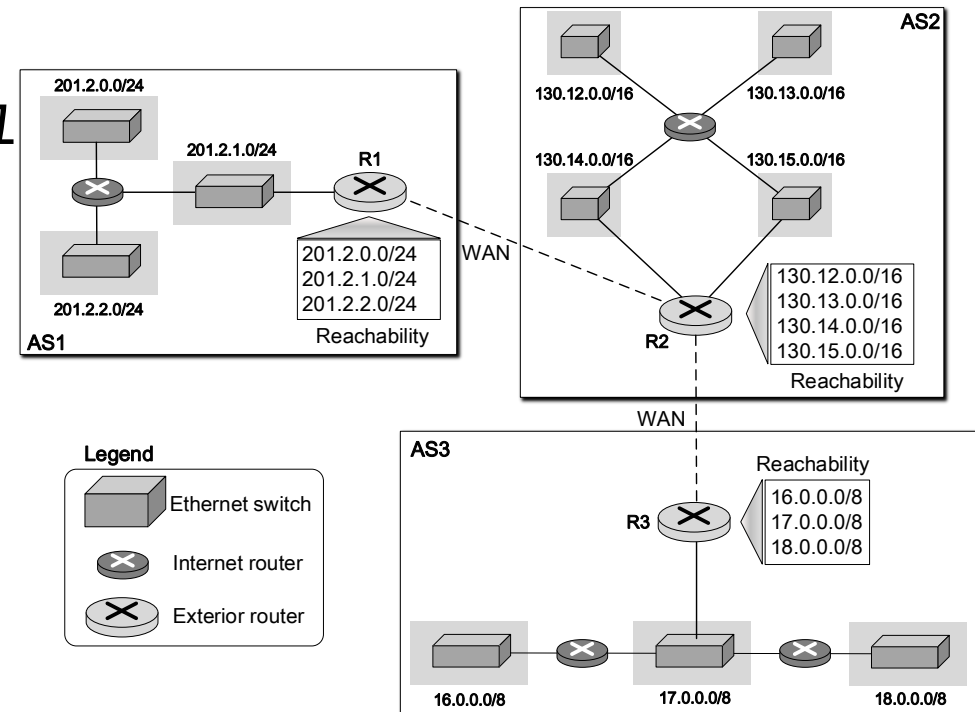
- 도메인 간 라우팅에 유용
- 원리는 거리 벡터 라우팅과 유사
 - 자신의 직접적인 이웃 라우터와 정보를 교환함
 - 이웃 라우터에게 받은 정보를 바탕으로 경로를 재구성
 - 전달하는 정보에서 차이점이 있음
 - 거리 벡터 라우팅은 목적지까지의 거리 값을 전달함
 - 경로 벡터 라우팅은 목적지까지 거쳐 가는 AS 전체 경로 목록을 전달함
- BGP (Border Gateway Protocol)로 구현

경로 벡터 라우팅

- 도달 가능성(Reachability)

- 개요

- 각 AS는 자신이 도달 가능한 네트워크 목록을 이웃 AS에게 광고해야 함
- 이웃 AS는 정보를 받아 목적지 네트워크로 가는 어떤 경로가 존재하는지 알게 됨
 - 도달 가능성 정보는 해당 AS에서 접근할 수 있는 네트워크의 목록을 의미함



경로 벡터 라우팅

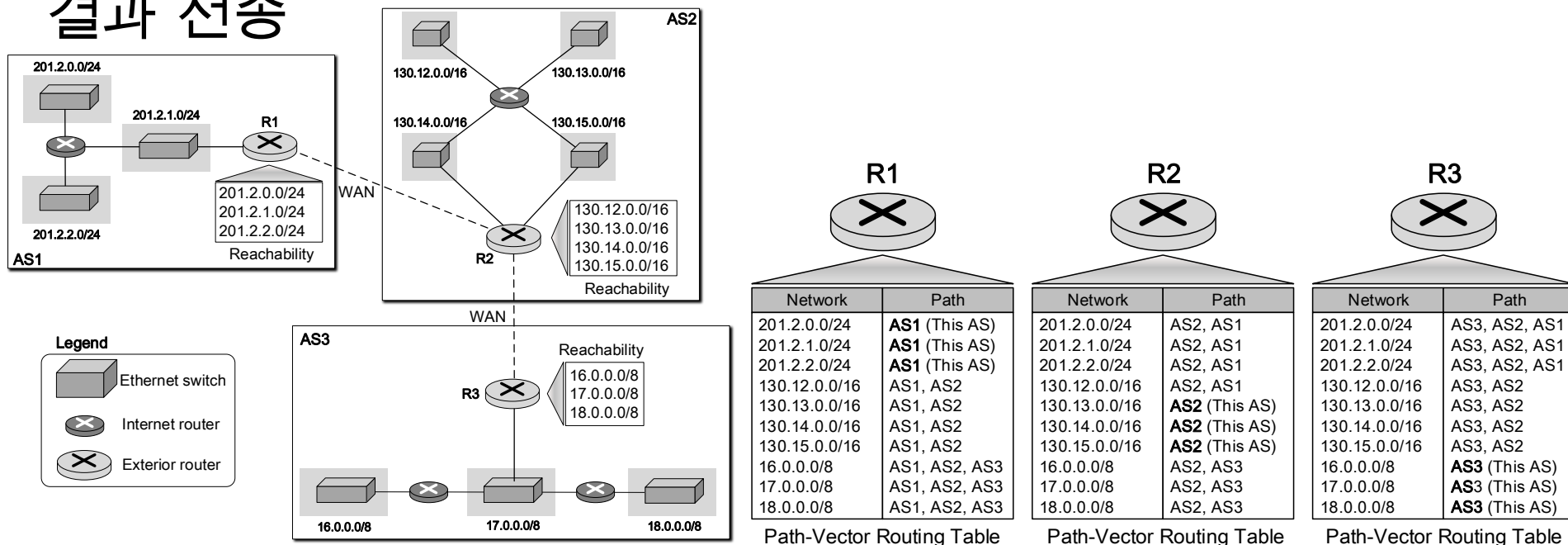
• 라우팅 테이블 (1/4)

• 개요

- AS가 도달 가능성 목록을 다른 것과 공유하게 되면 생성

• 동작 과정

- R1은 R2에 자신의 도달 가능성 목록을 전송할 수 있음
- R2는 자신의 도달 가능성 정보와 합친 후 라우터 R1과 R3에 결과 전송

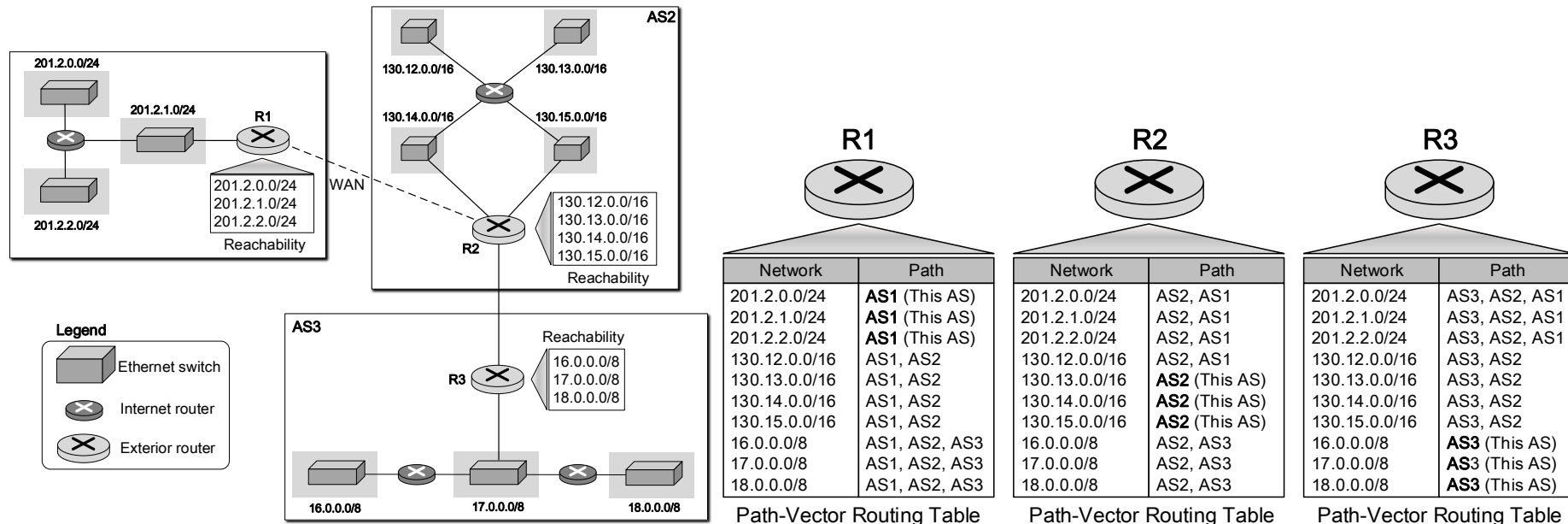


경로 벡터 라우팅

• 라우팅 테이블 (2/4)

• 동작 과정

- R3는 R2에 도달 가능성 정보 전달 후 자신의 라우팅 테이블 개선하고 각 라우터에 해당 정보를 재전파
- R1은 130.14.0.0/16 네트워크로 가는 패킷이 들어오면 목적지 네트워크에 도달하기 위해서 AS1에서 AS2로 가야 함을 알게 됨



경로 벡터 라우팅

- 라우팅 테이블 (3/4)

- 루프 방지

- 거리 벡터 라우팅에서 일어나는 불안정과 루프의 생성은 경로 벡터 라우팅에서 해결될 수 있음
 - 라우터가 메시지를 수신하면 목적지까지의 경로 목록에 해당 자율 시스템이 있는지를 검토
 - 만약 자율 시스템이 있다면 루프가 일어난 것이므로 메시지를 무시

- 정책 라우팅(Policy routing)

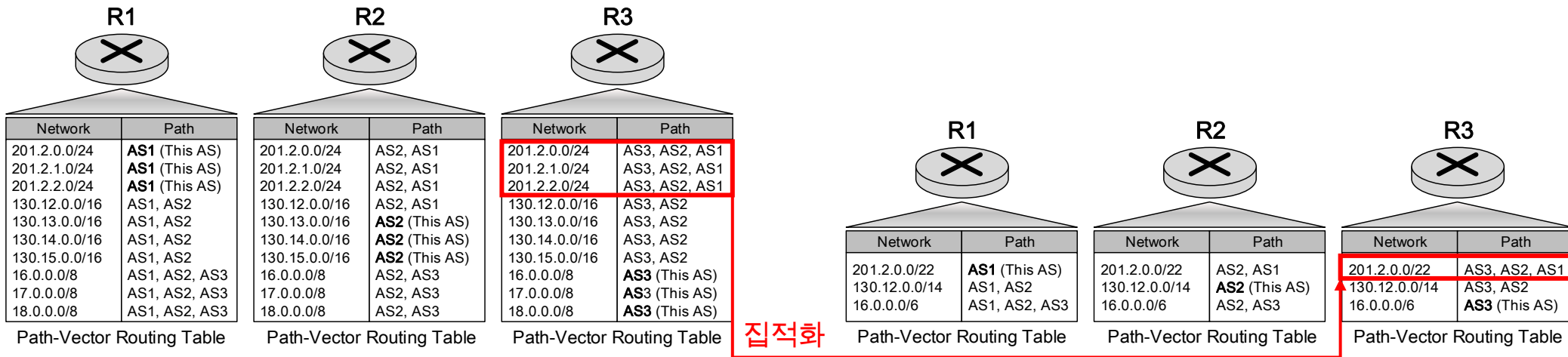
- 라우터가 메시지를 수신하면 경로를 검사하여 경로 상에 있는 자율 시스템들 중 하나가 정책을 위반하는지 판단
 - 위반되면 해당 경로에 대한 라우팅 테이블을 갱신하지도, 이웃에 전송하지도 않음

경로 벡터 라우팅

• 라우팅 테이블 (4/4)

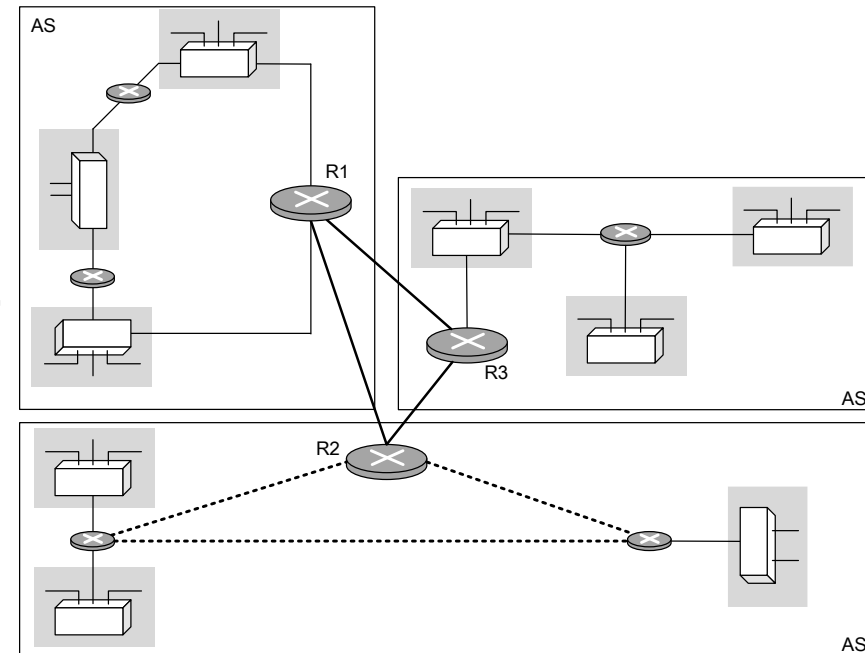
• 집적화 (Aggregation)

- 일반적으로 CIDR 표기가 가능하면 주소의 집적화를 지원
 - 경로 벡터 라우팅을 더 간단하게 만들어 주고 라우터 간의 정보 교환을 빠르게 개선



경로 벡터 라우팅

- 경계 게이트웨이 프로토콜(BGP, Border Gateway Protocol) (1/17)
- 정의
 - 자율 시스템 간의 라우팅 프로토콜
- 특징
 - 경로 벡터 프로토콜
 - 정책 기반 라우팅
 - TCP를 사용한 신뢰성 있는 연결



경로 벡터 라우팅

- BGP (2/17)

- 자율 시스템 유형 (1/2)

- 스템브 AS(Stub AS)

- 다른 자율 시스템과 단 하나의 연결을 가짐
 - 스템브 AS 내의 도메인 간 데이터 트래픽은 AS내에서 생성되거나 사라질 수 있음
 - 데이터 트래픽은 스템브 AS를 통해 지나갈 수 없음
 - 외부에서 들어온 트래픽은 스템브 AS에서 끝남
 - 스템브 AS 내부에서 생성된 트래픽만 외부로 나감
 - e.g., 작은 로컬 ISP

- 멀티홈 AS(Multihomed AS)

- 다른 자율 시스템과 하나 이상의 연결을 가짐
 - 데이터 트래픽의 송신자나 수신자만 될 수 있음
 - 하나의 자율 시스템에서 다른 시스템으로 지나가는 트래픽은 허용되지 않음
 - e.g., 국가 AS에 연결된 큰 조직

경로 벡터 라우팅

- BGP (3/17)
- 자율 시스템 유형 (2/2)
 - 경유 AS
 - 지나가는 트래픽을 허용하는 멀티홉 AS
 - e.g., 국가 혹은 국제 인터넷 백본
- CIDR (Classless Inter-Domain Routing)
 - 고정된 IP대신 네트워크 주소 범위를 유연하게 할당하여 IP 주소 할당 효율성을 높이고 인터넷 라우팅 테이블 크기를 줄이는 방식

경로 벡터 라우팅

- BGP (4/17)

- 경로 속성

- 잘 알려진 속성

- 모든 BGP 라우터가 반드시 인식해야만 하는 속성
 - 필수적인 속성
 - 경로를 기술하는데 반드시 필요한 속성
 - 임의(Discretionary) 속성
 - 각 라우터가 반드시 인식할 수 있어야 하나 모든 갱신 메시지에 포함되지 않아도 되는 속성

- 선택적인 속성

- 모든 라우터에서 인식될 필요는 없는 속성
 - 선택적 천이(Transitive) 속성
 - 라우터가 속성을 지원하지 않더라도 다음 라우터로 전달되어야만 하는 속성
 - 선택적 비천이(Nontransitive) 속성
 - 수신한 라우터가 이 속성을 지원하지 않으면 버려지는 속성

경로 벡터 라우팅

- BGP (5/17)

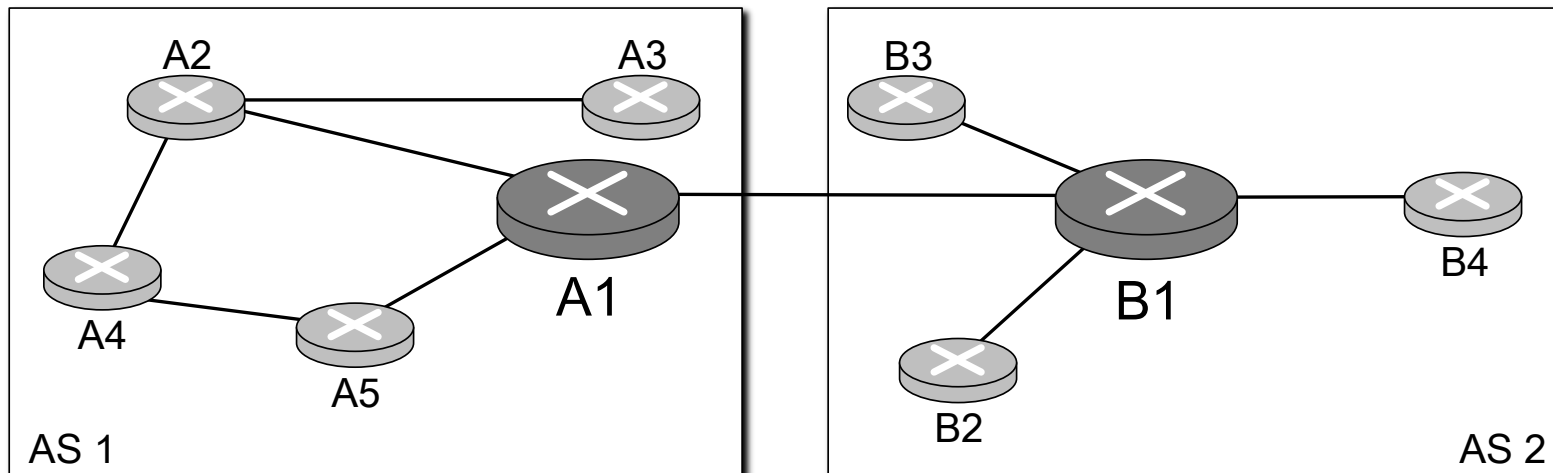
- 세션 (1/2)

- 정의

- 두 라우터 간 정보를 교환하기 위해 BGP 라우터 간 설정되는 연결

- 특징

- 신뢰성 있는 환경을 만들기 위해 TCP 서비스를 사용함
 - BGP에서 TCP 연결이 수립되면, 예외 사항이 발생하기 전까지 오랜 시간 세션이 유지됨
 - 이러한 이유로 반영구적 연결로도 불림



경로 벡터 라우팅

- BGP (6/17)

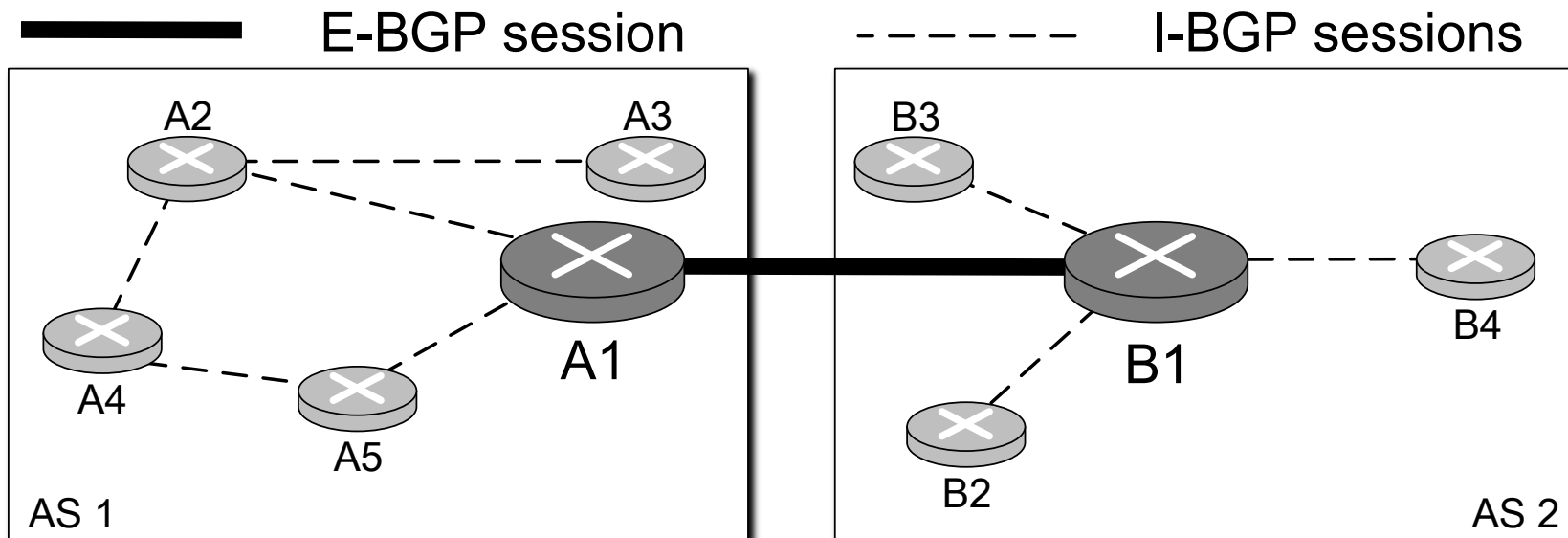
- 세션 (2/2)

- 외부(External) 세션

- 서로 다른 자율 시스템에 속하는 두 스피커 노드들 간에 정보 교환을 위해 사용

- 내부(Internal) 세션

- 자율 시스템 내의 두 라우터 간에 정보 교환을 위해 사용

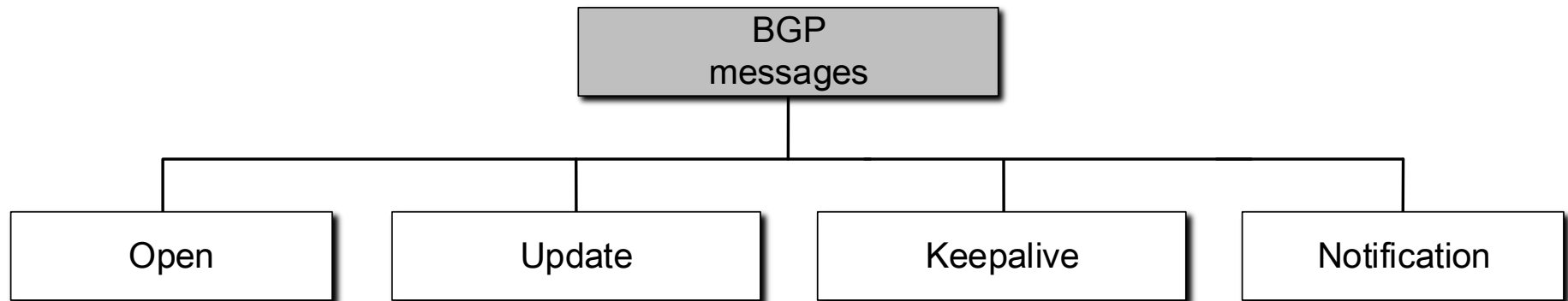


경로 벡터 라우팅

- BGP (7/17)

- 패킷 종류

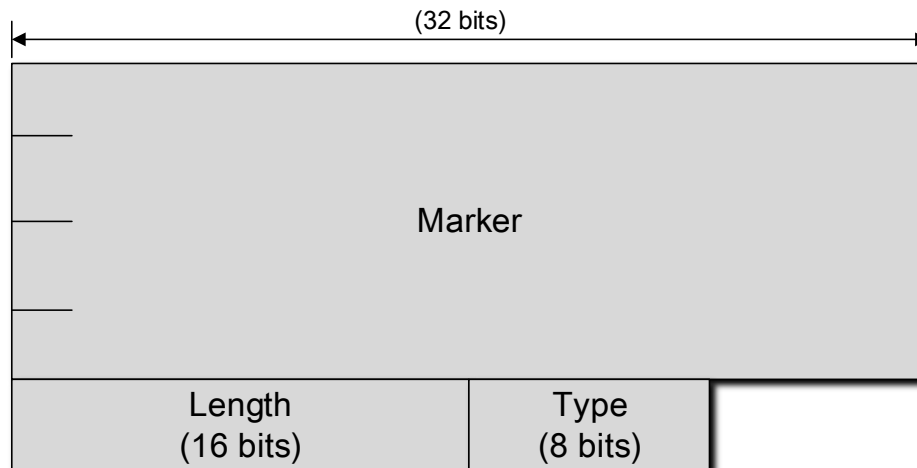
- 개방(Open)
- 갱신(Update)
- 킵얼라이브(Keepalive)
- 통지(Notification)



경로 벡터 라우팅

- BGP (8/17)
 - 패킷 형식 - 공통 헤더

필드	크기	역할
표시기 (Marker)	16 바이트	인증을 위해 예약
길이 (Length)	2 바이트	헤더를 포함한 저네 메시지의 길이를 표시
유형 (Type)	1 바이트	패킷의 유형을 1부터 4까지의 값으로 표시

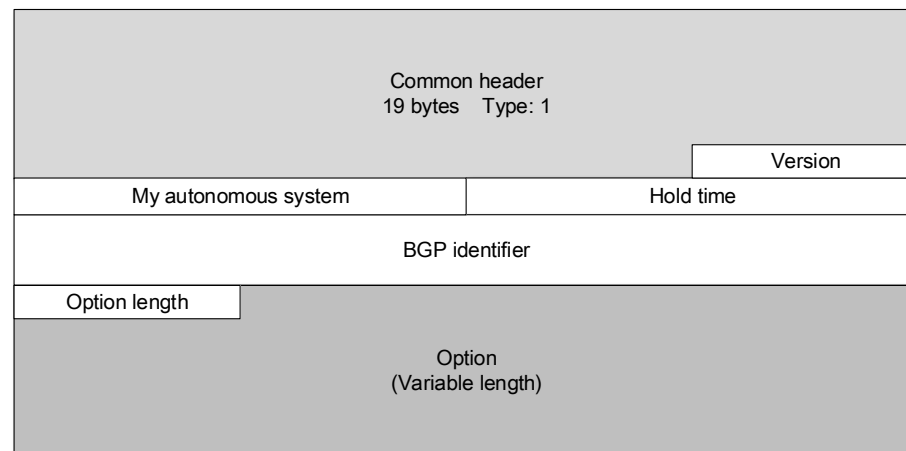


경로 벡터 라우팅

- BGP (9/17)

- 패킷 형식 - 개방 메시지(Open message) (1/3)

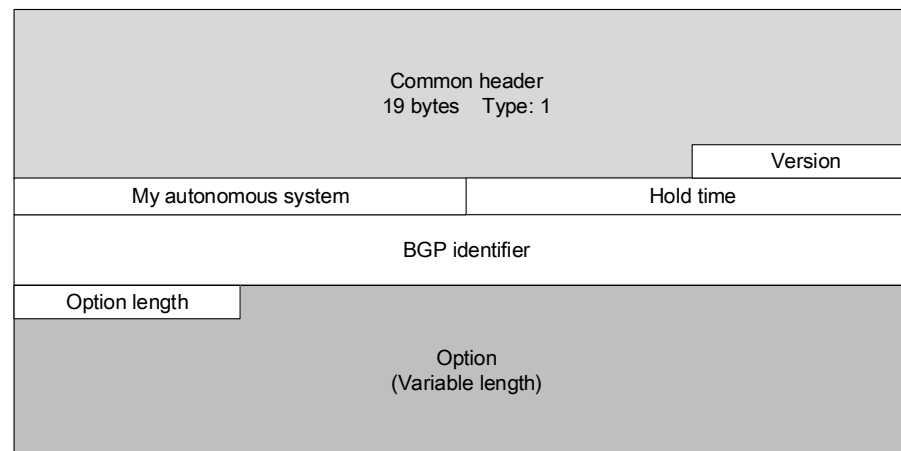
- BGP가 동작중인 라우터가 이웃 관계를 생성하기 위해서는 이웃과 TCP 연결을 열고 개방 메시지를 전송
- 이웃이 관계를 받아들이면 두 라우터 간에 이웃 관계가 성립되었다는 뜻으로 킵얼라이브 메시지를 전송



경로 벡터 라우팅

- BGP (10/17)
- 패킷 형식 - 개방 메시지 (2/3)

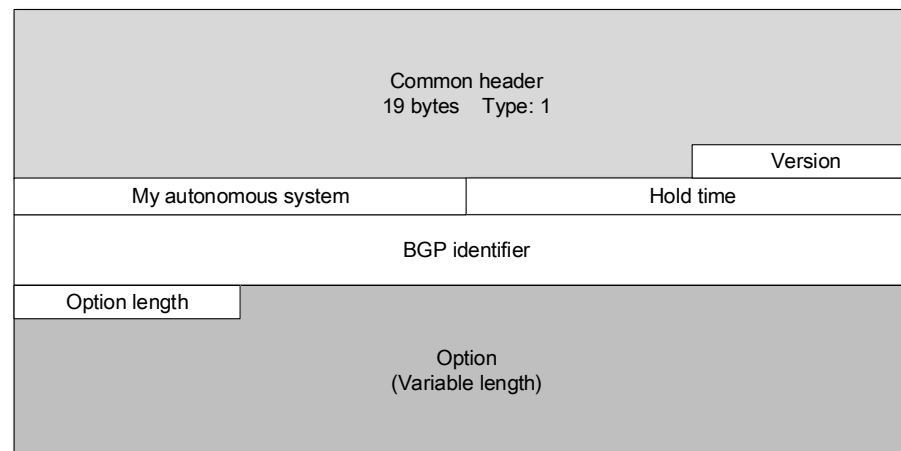
필드	크기	역할
버전 (Version)	1 바이트	BGP의 버전을 표시, 현재 값은 4
자신의 자율 시스템 (My autonomous system)	2 바이트	자율 시스템 번호를 표시
유지 시간 (Hold time)	2 바이트	상대방으로부터 킵얼라이브나 갱신 메시지를 수신하기 전까지 경과될 수 있는 시간을 초 단위로 표시



경로 벡터 라우팅

- BGP (11/17)
- 패킷 형식 - 개방 메시지 (3/3)

필드	크기	역할
BGP 식별자 (BGP Identifier)	4 바이트	개방 메시지를 전송한 라우터의 IP 주소 표시
선택 매개 변수 길이 (Option length)	1 바이트	전체 선택 매개 변수들의 길이를 표시 (e.g., 인증, 멀티프로토콜 확장 등)
선택 매개 변수들	변동	각 선택 매개 변수들은 매개변수 길이, 매개변수 값의 서브 필드를 보유 (현재까지 표준화된 선택 매개변수는 인증 뿐임)

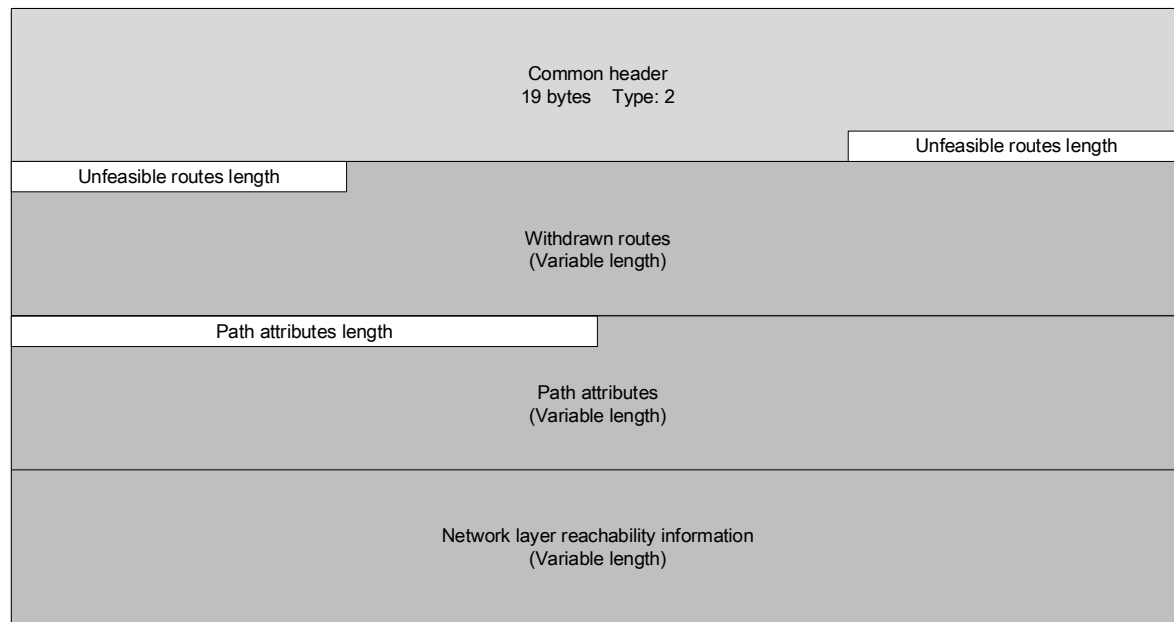


경로 벡터 라우팅

- BGP (12/17)

- 패킷 형식 - 갱신 메시지(Update message) (1/3)

- 라우터로 하여금 전에 광고된 목적지를 취소하거나 새로운 목적지로의 경로를 알리는 데 사용
- BGP는 전에 광고된 여러 목적지들을 취소할 수 있으나 새로운 목적지를 광고할 경우 하나의 갱신 메시지를 통해 한 목적지만 광고 가능

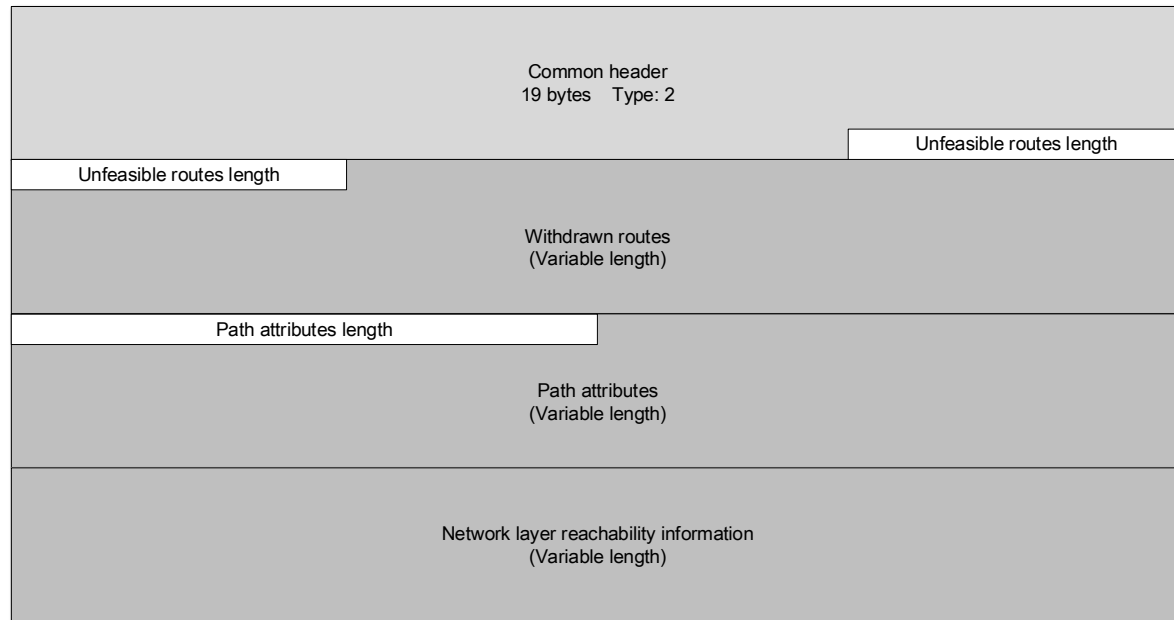


경로 벡터 라우팅

- BGP (13/17)

- 패킷 형식 - 갱신 메시지 (2/3)

필드	크기	역할
불가능(Unfeasible) 경로 길이	2 바이트	다음 필드의 길이 정의
취소된(Withdraw) 경로	변동	전에 광고된 목록 중에서 삭제되어야 하는 모든 경로를 나열
경로 속성 길이 (Path attributes length)	2 바이트	다음 필드의 길이 정의

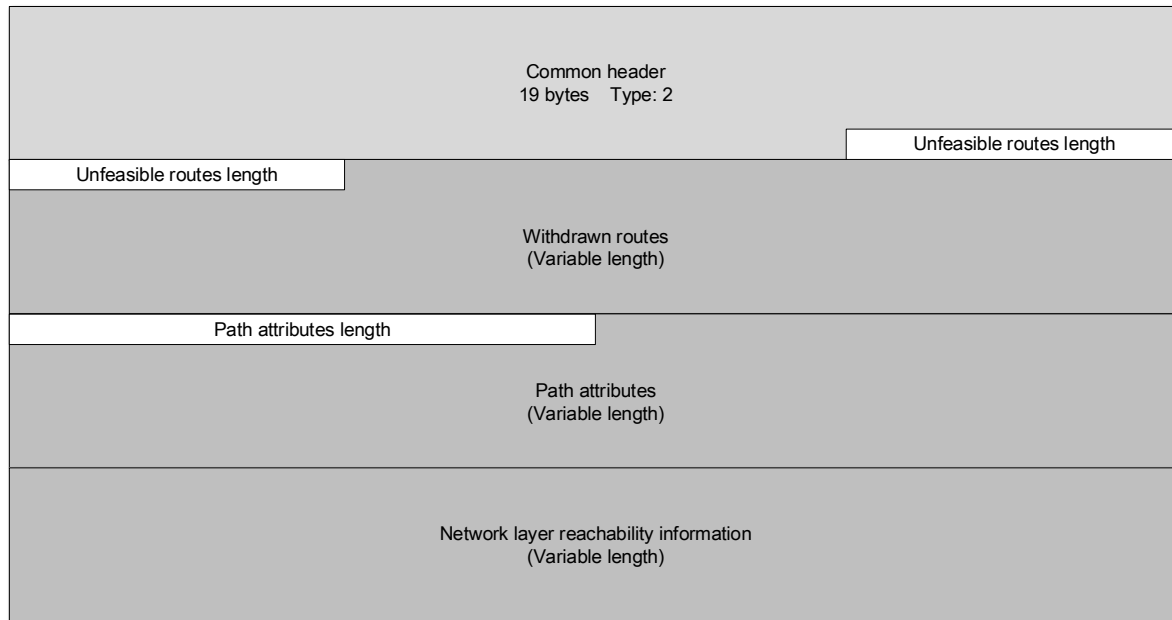


경로 벡터 라우팅

- BGP (14/17)

- 패킷 형식 - 갱신 메시지 (3/3)

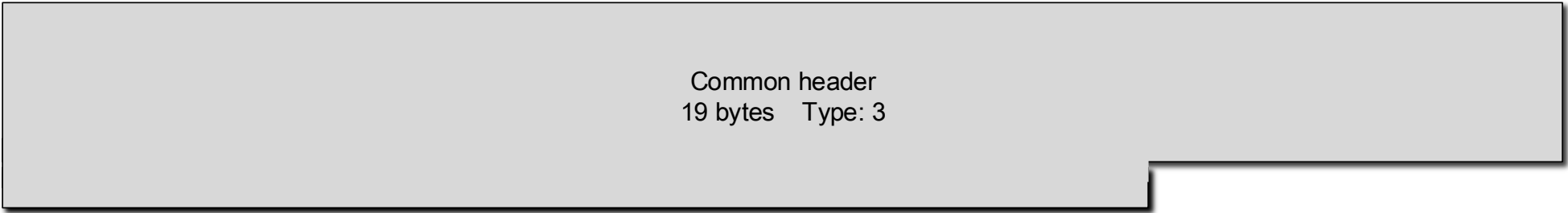
필드	크기	역할
경로 속성 (Path attributes)	변동	메시지를 통해 광고되는 도달 가능한 네트워크까지의 경로에 대한 속성 표시
네트워크 계층 도달 가능 정보 (NLRI, Network Layer Reachability Information)	변동	메시지에 의해 실제로 광고되어지는 네트워크 표시 길이 필드와 IP주소 프리픽스를 보유



경로 벡터 라우팅

- BGP (15/17)

- 패킷 형식 - 킵얼라이브 메시지(Keepalive message)
 - 상대방에게 자신들이 살아있음을 알리기 위해 유지 시간이 만료되기 전에 정기적으로 교환하는 메시지
 - 공통 헤더만으로 구성



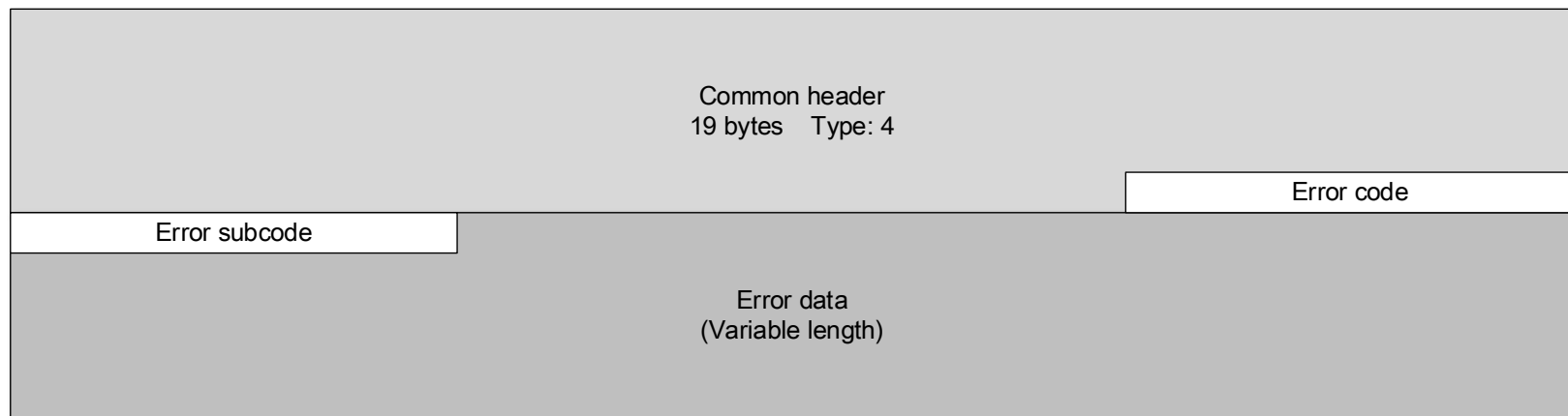
Common header
19 bytes Type: 3

경로 벡터 라우팅

- BGP (16/17)

- 패킷 형식 - 통지 메시지(Notification message)
 - 오류 상황이 감지되거나 연결을 닫기 원할 때 라우터에 의해 전송되는 메시지

필드	크기	역할
오류 코드(Error code)	1 바이트	오류의 분류를 표시
오류 서브 코드 (Error subcode)	1 바이트	각 분류에서의 오류 유형 표시
오류 데이터(Error data)	변동	오류에 대한 진단 정보를 제공

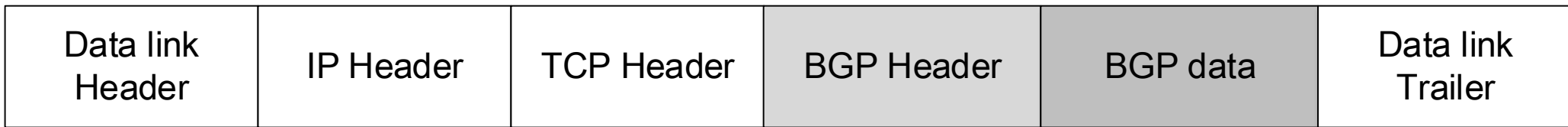


경로 벡터 라우팅

- BGP (17/17)

- 캡슐화

- 잘 알려진 포트 179를 사용하여 TCP 세그먼트에 캡슐화 됨
 - 오류 제어나 흐름 제어가 필요하지 않음
 - 종료(Cease) 유형의 통지 메시지가 전송될 때까지 갱신, 킵얼라이브, 통지 메시지의 교환을 지속함



목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

유니캐스팅과 멀티캐스팅

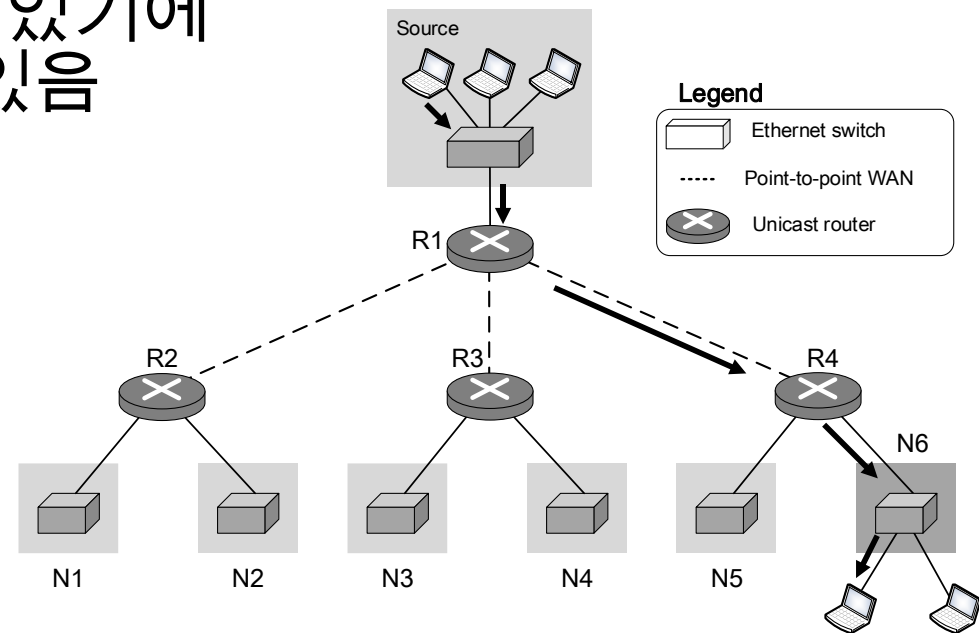
- 유니캐스팅(Unicasting)

- 정의

- 송신자와 목적지 간에 일대일로 메시지를 전송하는 방식

- 특징

- 라우터가 수신한 패킷을 하나의 인터페이스로만 전달
- 모든 호스트가 패킷을 볼 수 있기에 브로드캐스트처럼 보일 수 있음



유니캐스팅과 멀티캐스팅

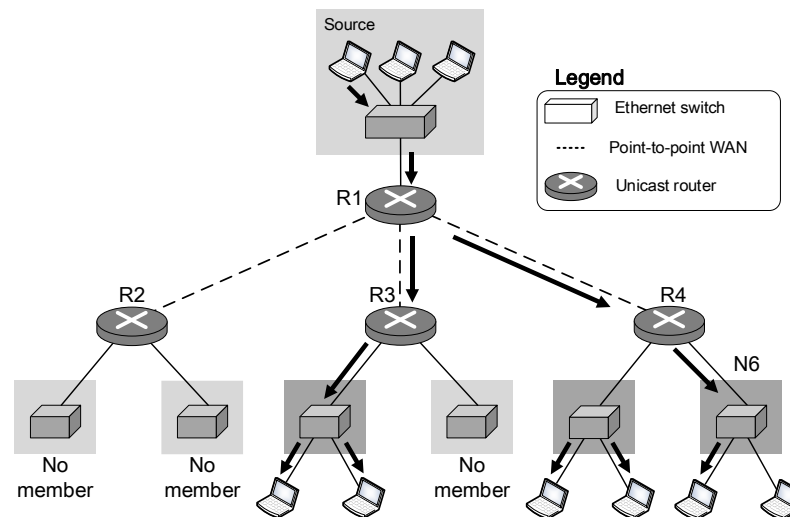
- 멀티캐스팅(Multicasting)

- 정의

- 하나의 송신자와 다수의 목적지가 있는 통신

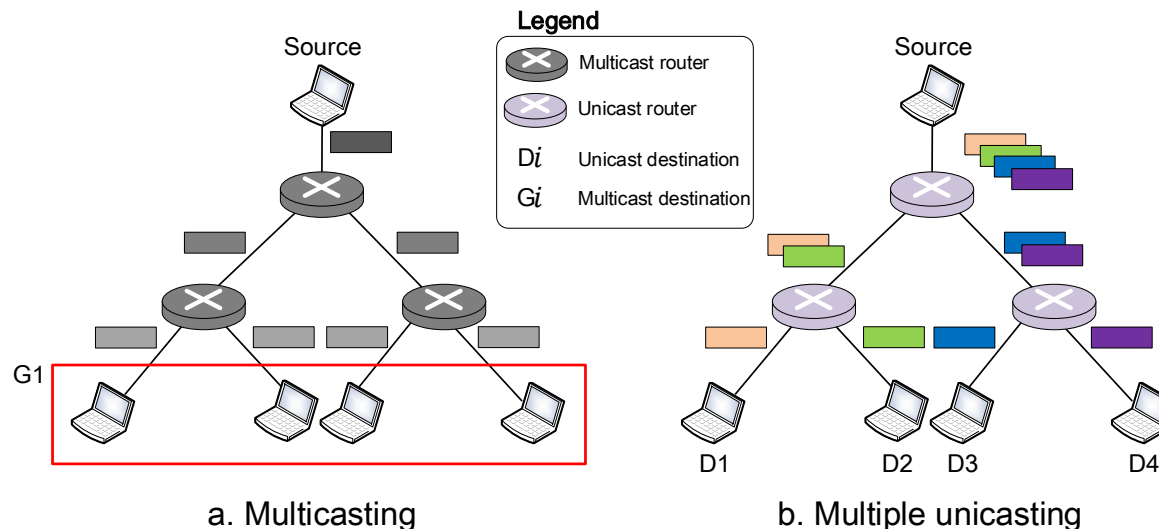
- 특징

- 라우터가 수신한 패킷을 여러 인터페이스를 통해 전달할 수 있음
 - 실시간 스트리밍, 원격 강의 등에서 사용됨



유니캐스팅과 멀티캐스팅

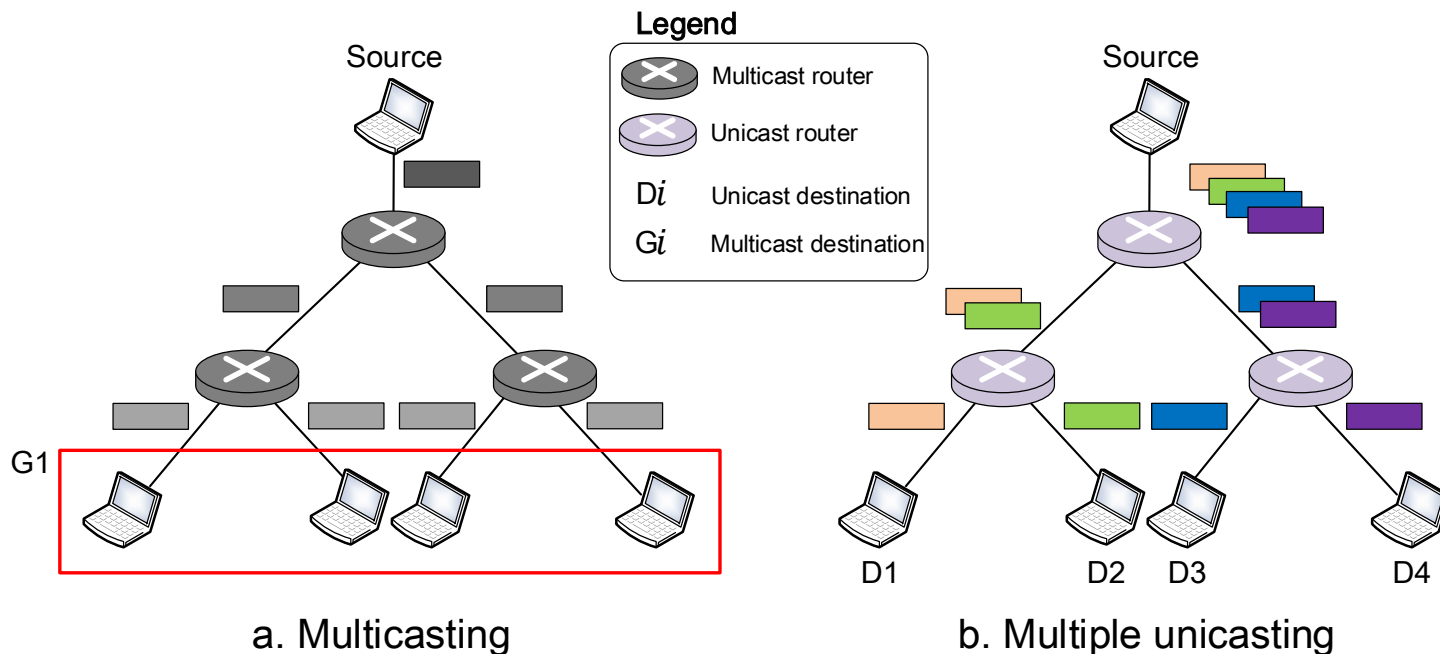
- 유니캐스팅과 멀티캐스팅의 차이점
 - 멀티캐스팅은 송신자에서 하나의 패킷으로 시작되어 라우터에서 필요에 의해 복사됨
 - 목적지 주소는 모든 중복된 패킷들에게 동일하게 부여됨
 - 임의의 라우터 간에는 하나의 패킷 사본만이 지나감
 - 복수의 유니캐스팅은 송신자로부터 다수의 패킷이 전송됨
 - 만약 목적지가 4개라면, 서로 다른 4개의 유니캐스트 목적지 주소를 가지는 4개의 패킷을 전송해야 함



유니캐스팅과 멀티캐스팅

- 복수의 유니캐스팅

- 복수의 유니캐스팅은 멀티캐스팅에 비해 비효율적임
 - 몇 링크는 다수의 사본들을 처리해야만 하기에 멀티캐스팅에 비해 많은 대역폭을 사용
 - 패킷들이 송신자에서 복사되어야 하기 때문에 각각의 패킷들이 상대적인 지연 시간을 가짐



유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (1/10)
 - 정의
 - 멀티캐스트 그룹에 가입한 호스트 그룹에 대한 목적지 주소
 - 특징
 - 멀티캐스트 주소를 목적지로 사용한 패킷은 수신자의 필터링 제약에 의한 경우를 제외하고는 모든 그룹 멤버에 전달될 수 있음

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (2/10)

- IPv4에서의 멀티캐스트 주소 (1/5)

- 멀티캐스팅에서 할당된 블록은 224.0.0.0/4임
 - 224.0.0.0에서 239.255.255.255까지 2^{28} 개의 주소가 있음

CIDR	Range	Assignment
224.0.0.0/24	224.0.0.0 → 224.0.0.255	Local Network Control Block
224.0.1.0/24	224.0.1.0 → 224.0.1.255	Internetwork Control Block
	224.0.2.0 → 224.0.255.255	AD HOC Block
224.1.0.0/16	224.1.0.0 → 224.1.255.255	ST Multicast Group Block
224.2.0.0/16	224.2.0.0 → 224.2.255.255	SDP/SAP Block
	224.3.0.0 → 231.255.255.255	Reserved
232.0.0.0/8	232.0.0.0 → 224.255.255.255	Source Specific Multicast (SSM)
233.0.0.0/8	233.0.0.0 → 233.255.255.255	GLOP Block
	234.0.0.0 → 238.255.255.255	Reserved
239.0.0.0/8	239.0.0.0 → 239.255.255.255	Administratively Scoped Block

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (3/10)

- IPv4에서의 멀티캐스트 주소 (2/5)

- 로컬 망 제어 블록(Local Network Control Block)

- 이 블록에 있는 주소들은 프로토콜 제어 트래픽으로 사용됨
 - 멀티캐스트 통신을 위해서는 사용되지 않음

- 이 구간의 주소들을 목적지로 사용하는 IP 패킷은 TTL (Time to Live) 값을 1로 설정해야 함

- 라우터로 하여금 패킷들을 전달하지 않도록 하기 위함

Address	Assignment
224.0.0.0	Base address (reserved)
224.0.0.1	All systems (hosts or routers) on this network
224.0.0.2	All routers on this network
224.0.0.4	DVMRP routers
224.0.0.5	OSPF routers
224.0.0.7	ST (stream) routers
224.0.0.8	ST (stream) hosts
224.0.0.9	RIP2 routers

Address	Assignment
224.0.0.10	IGRP routers
224.0.0.11	Mobile Agents
224.0.0.12	DHCP servers
224.0.0.13	PIM routers
224.0.0.14	RSVP encapsulation
224.0.0.15	CBT routers
224.0.0.22	IGMPv3

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (4/10)
 - IPv4에서의 멀티캐스트 주소 (3/5)
 - 망 간 제어 블록(Internetwork Control Block)
 - 로컬 네트워크 내 제어용 블록
 - 라우터들이 서로 제어 메시지를 주고 받는 용도
 - AD-HOC 블록
 - 임시 멀티캐스트 실험이나 임시 그룹 통신을 위해 마련된 블록
 - 주로 중앙 서버가 없는 P2P (Peer-to-Peer) 방식으로 동작
 - 블록의 크기가 100으로, CIDR 표기가 가능하기 위해선 IP 주소 블록의 크기가 2^x (x 는 양수) 이어야 해서 여러 개의 CIDR 블록을 조합해 표현해야 함

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (5/10)
- IPv4에서의 멀티캐스트 주소 (4/5)
 - 스트림 멀티캐스트 그룹 블록
 - 스트림 멀티미디어를 위한 블록
 - 주로 스트리밍이나 실시간 데이터 전송용 서비스 그룹에 사용
 - SAP, SDP 블록
 - 세션 선언 프로토콜 및 세션 디렉터리 프로토콜(RFC 2974)에 사용하는 블록
 - SSM (Source Specific Multicast) 블록
 - 특정 송신자로부터만 트래픽을 받도록 하는 블록

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (6/10)
 - IPv4에서의 멀티캐스트 주소 (5/5)
 - GLOP 블록
 - 특정 기관이 전 세계적으로 고유한 멀티캐스트 IP 주소 범위를 가질 수 있도록 하는 블록
 - 글로벌 콘텐츠 전송을 위해 각 조직이 자율적으로 주소를 사용할 수 있게 설정
 - 관리 범주 블록(Administratively Scoped Block)
 - 특정 조직이나 사이트 내부에서만 사용하도록 약속된 블록
 - 해당 대역을 사용하는 트래픽은 외부로 유출되거나 외부에서 트래픽이 유입되는 일이 없도록 설정

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (7/10)

- 주소 선택

- 제한된 그룹

- 특정 조직이나 기관에 할당되어 관리자의 승인이 필요한 멀티캐스트 그룹
 - GLOP 블록, 관리 범주 블록이 해당

- 대규모 그룹

- 별도의 승인 없이 누구나 사용할 수 있는 멀티캐스트 그룹
 - 송신자의 IP 주소 자체가 채널을 구별하는 고유 식별자가 되기 때문에, 같은 그룹 주소를 사용하더라도 충돌이 발생하지 않음
 - SSM 블록이 해당

유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (8/10)

- 데이터링크 계층으로의 패킷 전달 (1/3)

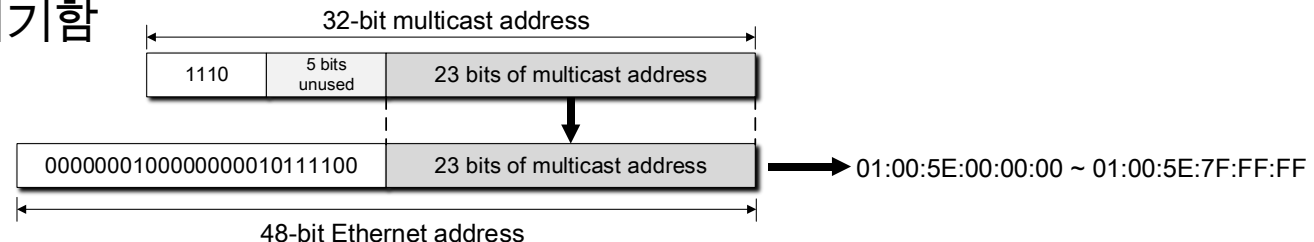
- 문제점

- IP 패킷은 멀티캐스트 주소를 가지므로 데이터링크 계층으로 패킷을 전달하기 위해 ARP 프로토콜로 물리 주소를 찾을 수 없음

- 해결 방안

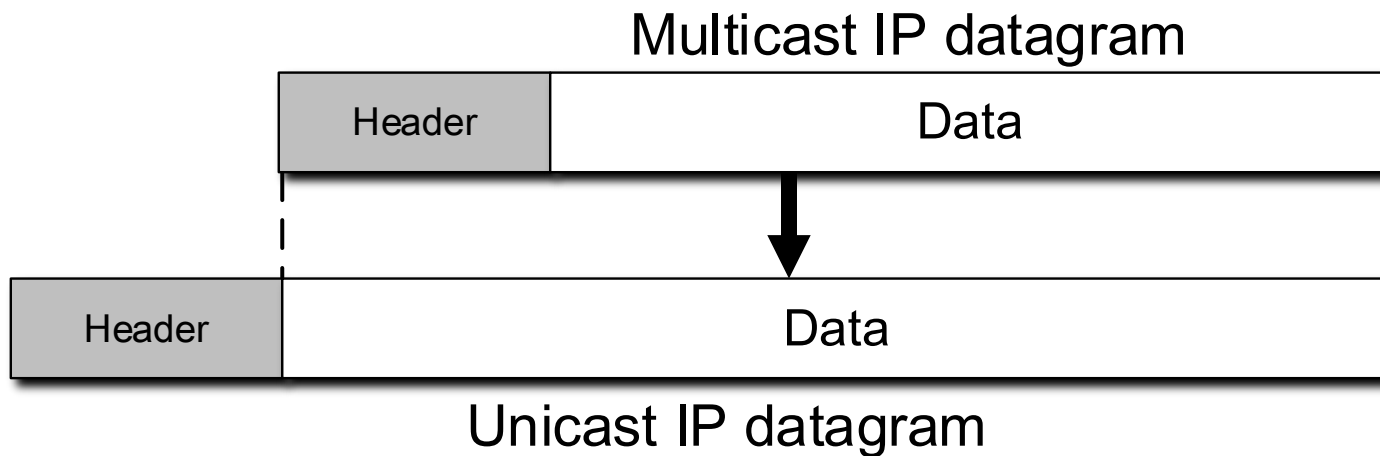
- 물리적 멀티캐스트 지원이 존재할 경우

- 이더넷 주소의 상위 25비트는 TCP/IP 프로토콜을 위한 것이고 나머지 23비트는 그룹을 정의하기 위해 사용되는 것을 이용
 - IP 멀티캐스트 주소를 이더넷 주소로 변환하기 위해 멀티캐스트 라우터는 IP 주소에서 하위 23비트를 추출하여 이더넷의 멀티캐스트 물리 주소에 삽입
 - IP 주소의 그룹 식별자는 28비트이므로, 상위 5비트가 사용되지 않음
 - 호스트는 IP 주소를 점검해서 자신의 IP 주소에 해당하지 않는 패킷은 폐기함



유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (9/10)
- 데이터링크 계층으로의 패킷 전달 (2/3)
 - 해결 방안
 - 물리적 멀티캐스트 지원이 존재하지 않는 경우
 - 대부분의 WAN은 물리 멀티캐스트를 지원하지 않음
 - 터널링(Tunneling) 과정을이용
 - 멀티캐스트 패킷이 유니캐스트 패킷에 캡슐화되어 전달되는 과정



유니캐스팅과 멀티캐스팅

- 멀티캐스트 주소 (10/10)
- 데이터링크 계층으로의 패킷 전달 (3/3)
- 예제 12.1

멀티캐스트 IP 주소 232.43.14.7을 이더넷 멀티캐스트 물리 주소로 변환하라.
이더넷 멀티캐스트 물리 주소의 범위 : 01:00:5E:00:00:00 ~ 01:00:5E:FF:FF:FF

- 풀이
 - IP 주소 오른쪽 23비트를 16진법으로 작성하면 2B:0E:07임
 - 232.43.14.7을 16진수로 작성하면 E8 2B 0E 07
 - 2진수로 나타내면 $(11101000\ 00101011\ 00001110\ 0000011)_2$
 - 오른쪽 23바이트는 $(0101011.00001110.0000111)_2$
 - 해당 값을 16진수로 변환하면 2B:0E:07
 - 이더넷 멀티캐스트 처음 주소인 01:00:5E:00:00:00에 더함
- ∴ 멀티캐스트 물리 주소는 01:00:5E:2B:0E:07

목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

IGMP

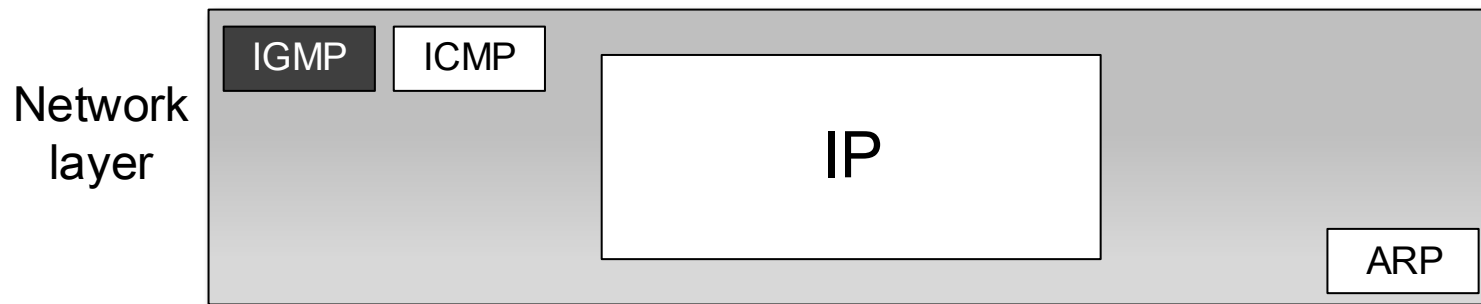
- 정의

- 멀티캐스트를 효율적으로 전달하기 위해 그룹 멤버에 대한 정보를 해독하고 수집하는 프로토콜

- 특징

- 3가지 버전을 보유

- 버전 1, 2는 ASM (Any Source Multicast)을 제공
 - 멀티캐스트 메시지가 어디에서 오든지 상관없이 그룹 멤버들이 수신
 - 버전 3는 SSM (Source Specific Multicast)을 제공
 - 미리 정해놓은 송신자로부터 오는 멀티캐스트 패킷만을 수신



IGMP

- 메시지 유형 (1/9)

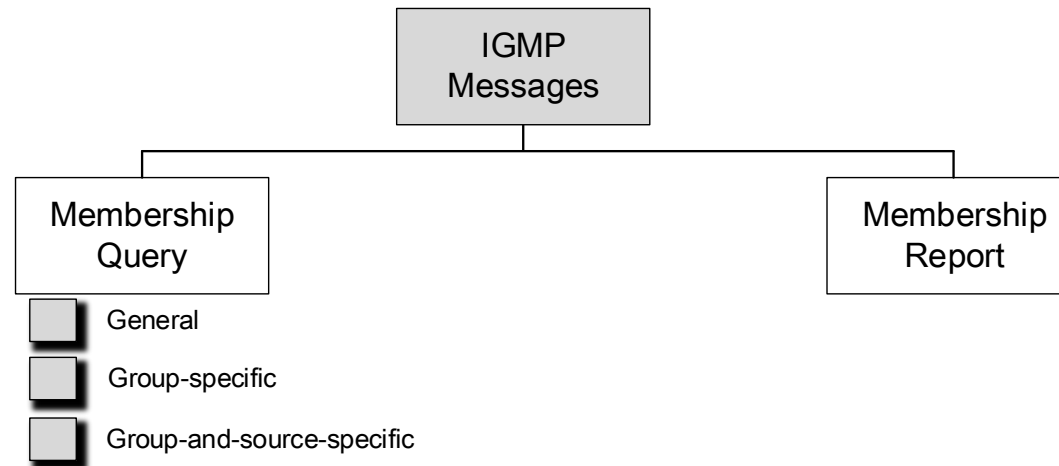
- 멤버십 질의(Membership Query)

- 정의

- 멀티캐스트 그룹에 참여하는 호스트가 있는지 주기적으로 확인 질문을 던지는 방법

- 종류

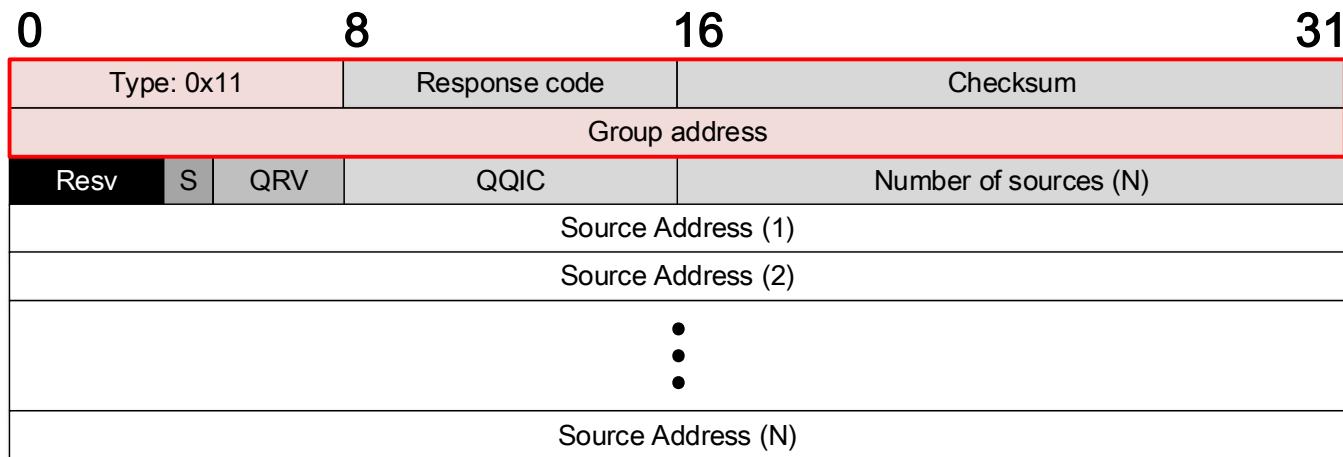
- 일반(General)
 - 그룹 지정 (Group-specific)
 - 그룹과 송신자 지정 (Group-and-source-specific)



IGMP

- 메시지 유형 (2/9)
- 멤버십 질의 메시지 형식 (1/4)

필드	크기	역할
유형 (Type)	8 비트	메시지 유형을 정의하며, 멤버십 질의 메시지의 경우 0x11 값을 가짐
최대 응답 코드 (Response code)	8 비트	질의가 응답되어야 하는 시간을 정의함
검사합 (Checksum)	16 비트	검사합 값을 저장하며, 전체 IGMP 메시지에 대해 계산됨
그룹 주소 (Group address)	32 비트	그룹 지정 및 그룹과 송신자 지정 질의 메시지를 보낼 때 IP 멀티캐스트 주소로 설정함



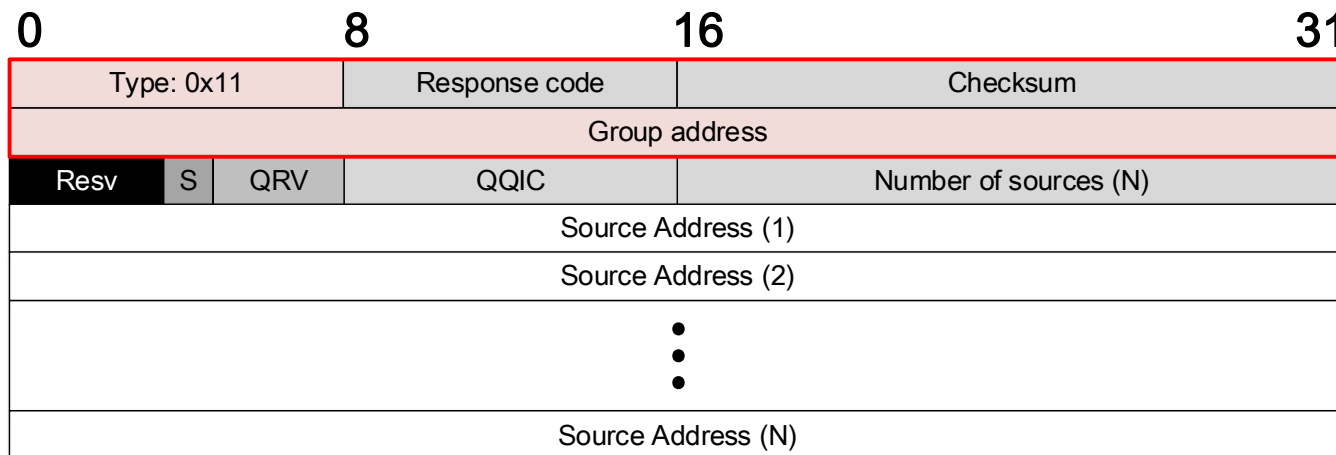
IGMP

• 메시지 유형 (3/9)

• 멤버십 질의 메시지 형식 (2/4)

*강인성 변수(Robustness Variable)
메시지 손실을 대비해 중복, 재전송 횟수를 결정하는 변수

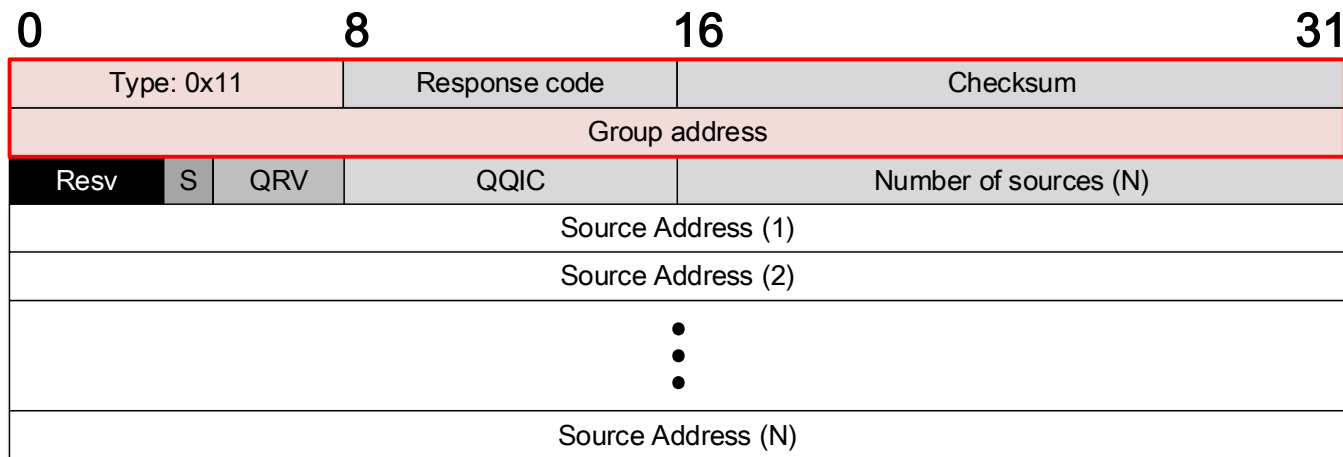
필드	크기	역할
Resv.	4 비트	향후를 위해 예약해둔 값
S. (Suppress)	1 비트	1로 설정되면 질의 메시지 수신자는 정규 타이머 갱신을 수행하지 않음
QRV (Querier's *Robustness Variable)	3 비트	네트워크 강인성을 모니터링하기 위해 사용됨
QQIC (Querier's Query Interval Code)	8 비트	질의 간격인 QQI 값을 계산하기 위해 사용됨



IGMP

- 메시지 유형 (4/9)
- 멤버십 질의 메시지 형식 (3/4)

필드	크기	역할
송신자 수 (Number of Sources)	16 비트	질의에 관여되는 32비트 유니캐스트 송신자 주소와 수를 정의함
송신자 주소 (Source Address)	32 비트	멀티캐스트 메시지의 송신자 주소를 나타냄



IGMP

• 메시지 유형 (5/9)

• 멤버십 질의 메시지 형식 (4/4)

• 일반 질의 메시지

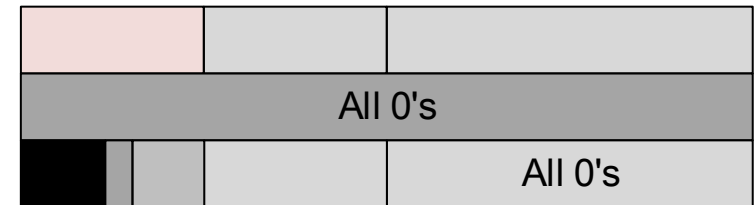
- 그룹 멤버십의 전체 목록을 보고하도록 각 이웃에게 요청

• 그룹 지정 질의 메시지

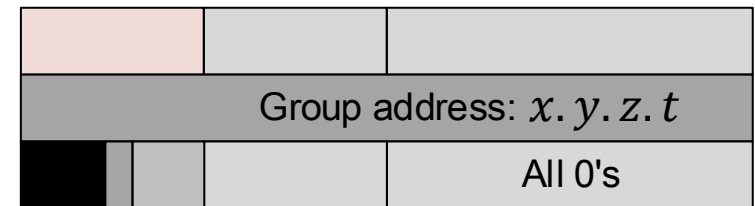
- 특정 멀티캐스트 그룹에 수신자가 있는지 확인
 - $x.y.z.t$ 는 멀티캐스트 그룹 주소

• 그룹과 송신자 지정 질의 메시지

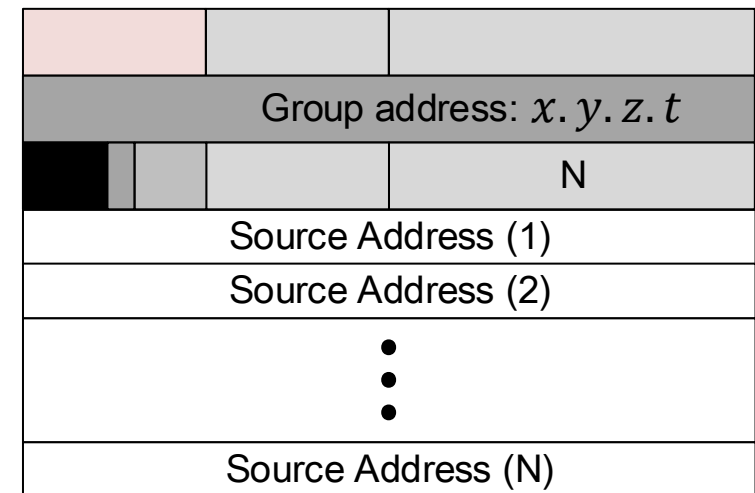
- 특정 멀티캐스트 그룹 안에서 특정 송신자로부터 오는 데이터를 받는 수신자가 있는지 확인



a. General



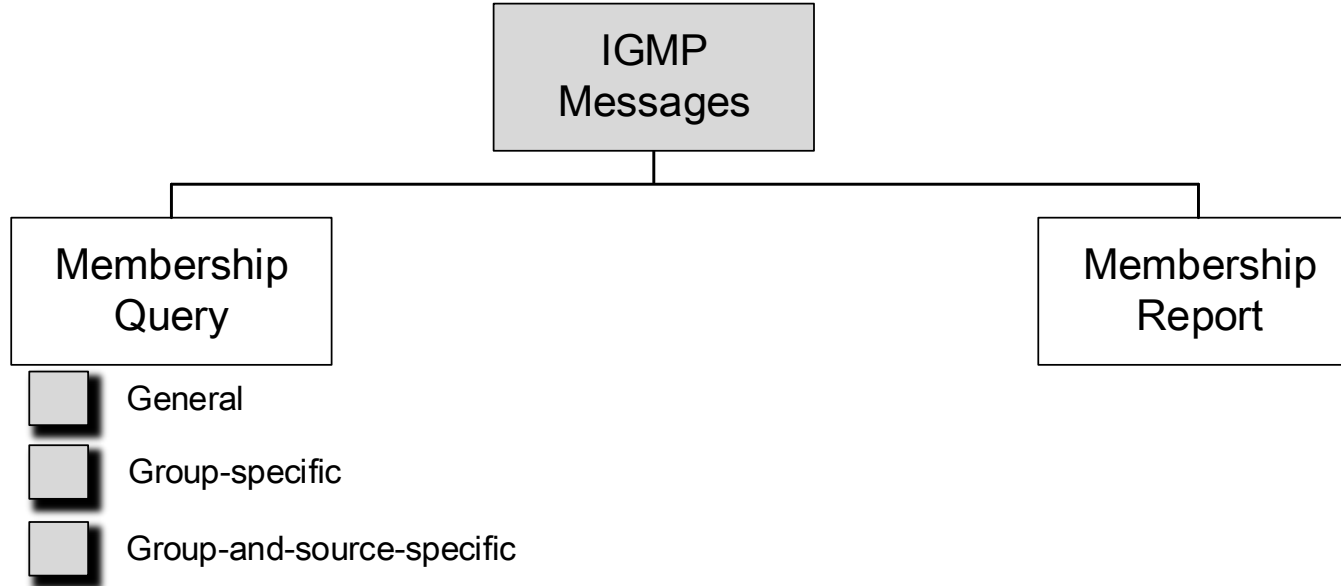
b. Group-specific



c. Group-and-source specific

IGMP

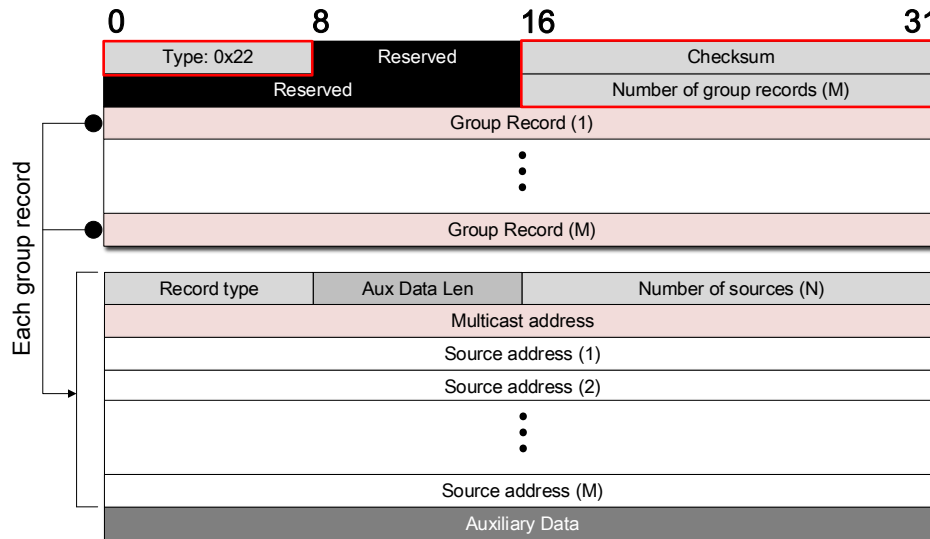
- 메시지 유형 (6/9)
- 멤버십 보고(Membership Report)
 - 정의
 - 호스트가 그룹 참여와 관련한 정보를 보고하는 방법



IGMP

- 메시지 유형 (7/9)
- 멤버십 보고 메시지 형식 (1/3)

필드	크기	역할
유형 (Type)	8 비트	메시지 유형을 정의하며, 멤버십 보고 메시지의 경우 0x22 값을 가짐
체크섬	16 비트	체크섬 값을 저장하며, 전체 IGMP 메시지에 대해 계산됨
그룹 레코드 수 (Number of group records)	16 비트	패킷에 전달되는 그룹 레코드의 수를 계산함

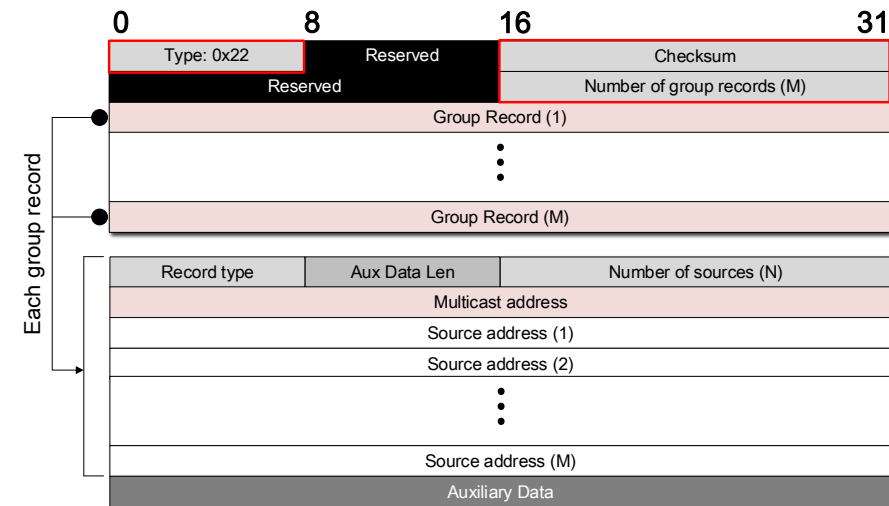


IGMP

- 메시지 유형 (8/9)
- 멤버십 보고 메시지 형식 (2/3)

필드	크기	역할
레코드 유형 (Record type)	8 비트	6개의 유형을 가짐
보조 데이터 길이 (Aux Data Len)	8 비트	각 그룹 레코드에 포함된 보조 데이터의 길이를 정의 32비트 단위를 가짐

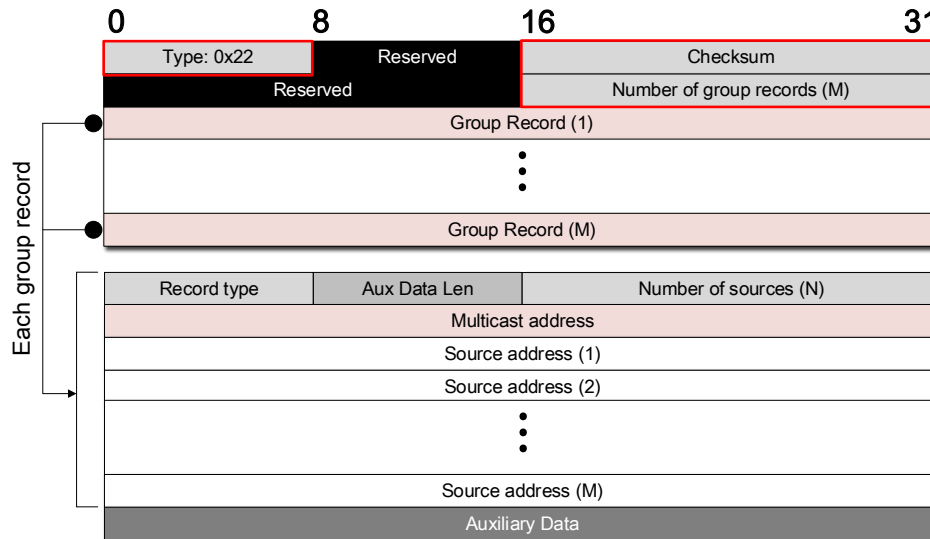
Category	Type	Type Value
Current-State-Record	Mode_Is_Include	1
	Mode_Is_Exclude	2
Filter-Mode-Change-Record	Change_To_Include_Mode	3
	Change_To_Exclude_Mode	4
Source-List-Change-Record	Allow_New_Sources	5
	Block_Old_Sources	6



IGMP

- 메시지 유형 (9/9)
- 멤버십 보고 메시지 형식 (3/3)

필드	크기	역할
송신자 수 (Number of sources)	16	보고에 포함되는 32비트 송신자 주소 수를 정의함
송신자 주소 (Source Address)	32	N개의 송신자 주소를 나타냄
보조 데이터 (Auxiliary Data)	N/A	보고 메시지에 포함될 수 있는 보조 데이터를 포함함 아직 정의되지 않음



IGMP

- 타이머와 변수 (1/2)

- 타이머

- 정의

- 질의 메시지에 대한 응답으로 보고 메시지를 보내기 전에 허용되는 최대 시간

- 특징

- 질의 메시지 헤더 안에 포함된 Max Resp Code 필드가 결정
 - Max Resp Code 값이 128미만이면 최대 응답 시간은 정수
 - Max Resp Code 값이 128이상이면 최대 응답 시간은 부동 소수

IGMP

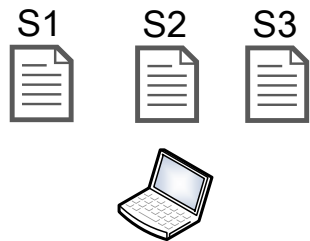
- 타이머와 변수 (2/2)
 - 질의자 강인성 변수(QRV, Querier's Robustness Variable)
 - 정의
 - 질의에 대한 응답 메시지를 얼마나 많이 전송해야 하는가를 나타내는 변수
 - 특징
 - 망에서 패킷 유실이 많으면 라우터는 QRV 값을 높게 설정
 - 호스트가 질의에 대해 응답을 보내는 횟수는 QRV-1임
 - 만약 라우터가 QRV 값을 5로 설정했다면 응답을 4번 보내야 함
 - 기본값은 2로 설정됨
 - 질의자 질의 간격(QQI, Querier's Query Interval)
 - 정의
 - 라우터가 멤버십 질의를 얼마나 자주 보낼지를 나타내는 시간 간격
 - 기본값은 125로 설정됨
 - 관리자가 망 내의 트래픽을 제어하기 위해 변경 가능

IGMP

• 호스트에 적용된 IGMP 프로토콜 (1/5)

• 소켓 상태

- 각 프로세스는 소켓으로부터 멀티캐스트 메시지 수신과 관련된 각 멀티캐스트 그룹에 관한 레코드를 가짐
 - Include 모드
 - 소켓으로부터 그룹 메시지를 수신하기를 원하는 송신자 주소의 목록을 가짐
 - Exclude 모드
 - 소켓을 통해 받아들이기를 원하지 않는 그룹 메시지의 송신자 주소를 가짐



Legend

S: Socket

a, b, ...: Source addresses

States Table

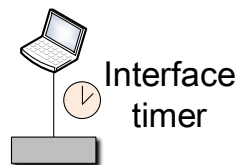
Socket	Multicast group	Filter	Source addresses
S1	226.14.5.2	Include	a, b, d, e
S2	226.14.5.2	Exclude	a, b, c
S2	228.24.21.4	Include	b, c, f
S3	226.14.5.2	Exclude	b, c, g
S3	228.24.21.4	Include	d, e, f

IGMP

• 호스트에 적용된 IGMP 프로토콜 (2/5)

• 인터페이스 상태

- 각 멀티캐스트 그룹에 대해 하나의 레코드만을 가짐
- 전체 상태에 대해 하나의 인터페이스 타이머와 각 레코드에 대해 하나 씩의 타이머를 가짐
- 레코드가 두 개 이상의 서로 다른 자원을 가진다면 합치기 위해 두 가지 법칙을 따름
 - 합칠 자원 중 하나라도 Exclude 모드를 가지는 경우
 - 최종 인터페이스 레코드는 Exclude 필터를 가짐
 - Exclude 필터를 가지는 모든 주소들에 대해 교집합 연산 수행한 후 결과에 Include 필터를 가지는 모든 주소 목록에 대해 차 연산 수행
 - 합칠 모든 레코드들이 Include 모드인 경우
 - 최종 인터페이스 레코드는 Include 필터 모드를 가지고 모든 주소 목록에 합 연산을 수행하여 송신자 주소의 목록을 찾음



Interface state

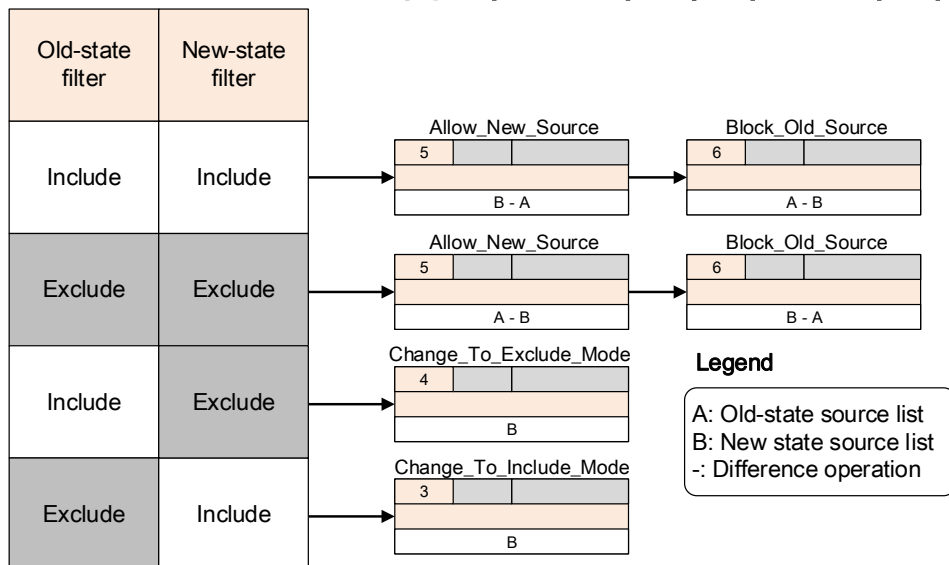
Multicast group	Group timer	Filter	Source addresses
226.14.5.2		Exclude	c
228.24.21.4		Include	b, c, d, e, f

IGMP

• 호스트에 적용된 IGMP 프로토콜 (3/5)

• 상태 변화를 확인한 후 전송

- 인터페이스 상태에 변화가 있을 경우 호스트는 레코드들을 사용하여 멤버십 보고 메시지를 보내야 함
- 이전 상태 및 새로운 상태 필터에 기반을 두어 4가지 서로 다른 변경이 있을 수 있음
 - Include-Include, Exclude-Exclude의 경우, 두 개의 그룹 레코드를 포함하고 나머지는 하나의 그룹 레코드를 포함하여 보고됨



Category	Type	Type Value
Current-State-Record	Mode_Is_Include	1
	Mode_Is_Exclude	2
Filter-Mode-Change-Record	Change_To_Include_Mode	3
	Change_To_Exclude_Mode	4
Source-List-Change-Record	Allow_New_Sources	5
	Block_Old_Sources	6

IGMP

- 호스트에 적용된 IGMP 프로토콜 (4/5)
 - 질의를 확인한 후 수신
 - 질의를 수신하면 즉각적으로 응답하지 않고, 수신된 질의의 유형에 따라 응답을 지연시킴
 - 유형
 - 일반 질의 수신
 - 호스트는 인터페이스 타이머를 계산된 지연 값으로 재설정함
 - 그룹 지정 질의 수신
 - 계산된 지연 시간과 기존 타이머의 잔여 시간 중 짧은 값으로 재설정함
 - 타이머가 동작하고 있지 않다면 잔여 시간을 무한대로 간주함
 - 그룹과 송신자 지정 질의 수신
 - 그룹 지정 질의 수신과 같은 행동을 수행하나 추가적으로 송신자의 목록이 지연 응답을 위해 기록됨

IGMP

• 호스트에 적용된 IGMP 프로토콜 (5/5)

• 타이머 만료

- 멤버십 보고 메시지는 타이머가 만료될 때 호스트에 의해 전송

• 유형

- 일반 질의를 수신한 후 설정된 인터페이스 타이머
 - 호스트는 인터페이스 상태에 있는 각 그룹 별로 하나의 Current-State-Record를 포함하는 멤버십 보고를 전송
- 그룹 지정 질의를 수신한 후 설정된 그룹 타이머
 - 호스트는 그룹이 여전히 활성화되어 있는 경우에만 특정 그룹에 대한 오직 하나의 Current-State-Record를 포함하는 멤버십 보고를 전송
- 그룹과 송신자 지정 질의를 수신한 후 설정된 그룹 타이머
 - 그룹 지정 질의와 같은 행동을 수행하나 레코드 유형과 송신자 목록은 그룹의 필터 모드에 의존

Category	Type	Type Value
Current-State-Record	Mode_Is_Include	1
	Mode_Is_Exclude	2

IGMP

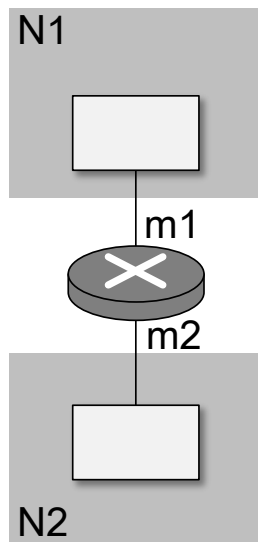
- 라우터에서의 적용

- 질의자의 상태

- 멀티캐스트 그룹에 대한 정보를 가진 테이블

- 멀티캐스트 주소, 그룹 타이머, 필터 모드 및 송신자 레코드로 구성됨
 - 각 송신자 코드는 송신자의 주소와 해당 타이머를 포함함

State for interface m1



Multicast group	Group timer	Filter	Source addresses
227.12.15.21		Exclude	(a,), (c,)
228.21.25.41		Include	(b,) , (d,) , (e,)

State for interface m2

Multicast group	Group timer	Filter	Source addresses
226.10.11.8		Exclude	(b,)
227.21.25.41		Include	(a,) , (b,) , (c,)
228.32.12.40		Include	(d,) , (e,) , (f,)

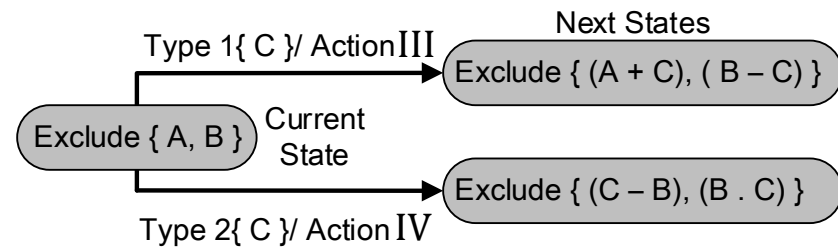
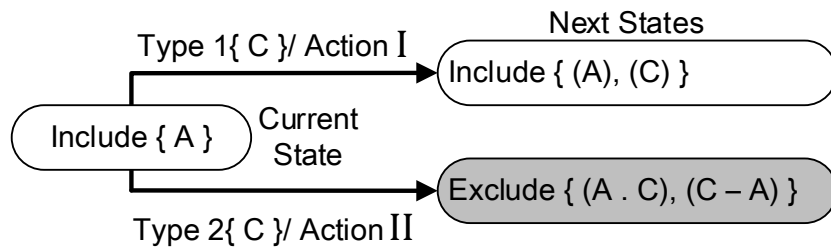
IGMP

• 호스트에 적용된 IGMP 프로토콜 (1/2)

• 멤버십 보고 수신에 대한 행동

• 일반 질의에 대한 응답으로 수신되는 보고

- 대부분 Current-State-Record를 포함
- 라우터의 각 레코드 상태는 현재 상태 및 보고에 도착한 레코드에 의존



	Group timer	Sources timers	Delete
I		Reset { C }	{ C - A }
II	Reset	Reset{ C - A }	
III		Reset { C }	
IV	Reset	Reset{ C - A - B }	{ A - C } & { B - C }

Category	Type	Type Value
Current-State-Record	Mode_is_Include	1
	Mode_is_Exclude	2
Filter-Mode-Change-Record	Change_To_Include_Mode	3
	Change_To_Exclude_Mode	4
Source-List-Change-Record	Allow_New_Sources	5
	Block_Old_Sources	6

IGMP

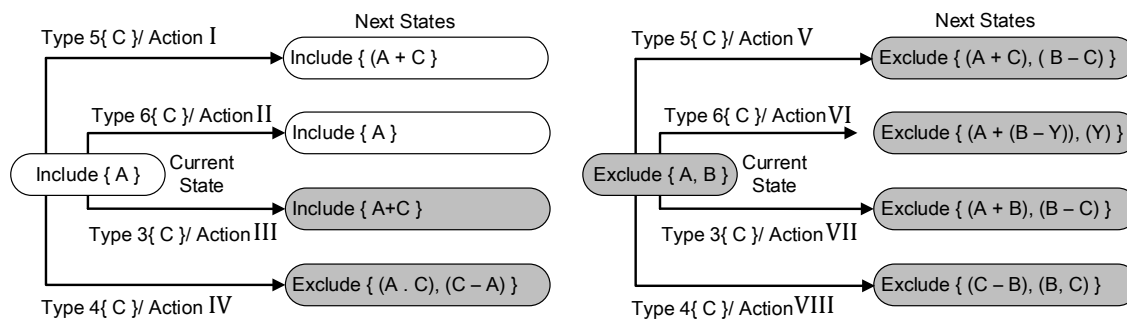
• 호스트에 적용된 IGMP 프로토콜 (2/2)

• 멤버십 보고 수신에 대한 행동

• 다른 질의들에 대한 응답으로 수신되는 보고

- 보통 Fliter_Mode_Change_Record 혹은 Source_List_Change을 포함

- 라우터에 있는 각 레코드의 다음 상태는 현재 상태와 도착한 보고에 포함된 레코드의 유형에 의존



	Group timer	Sources timers	Delete	Send G-S query	Send G-S-S query
I		Reset { C }			
II					Q - G - S{A, C}
III		Reset { C }			Q - G - S{A - C}
IV	Reset		{ A - C }		Q - G - S{A, C}
V		Reset { C }			
VI		Reset{ C - A - B }			Q - G - S{C - B}
VII		Reset { C }		Q - G	Q - G - S{A - C}
VIII	Reset		{ A - C } & { B - C }		Q - G - S{A - B}

Category	Type	Type Value
Current-State-Record	Mode_is_Include	1
	Mode_is_Exclude	2
Filter-Mode-Change-Record	Change_To_Include_Mode	3
	Change_To_Exclude_Mode	4
Source-List-Change-Record	Allow_New_Sources	5
	Block_Old_Sources	6

IGMP

- 전달(Forwarding)에서의 IGMP의 역할
- 패킷의 전달 여부를 멀티캐스트 라우터에 알려주기 위한 6가지 권장 사항이 IGMPv3에 근거함
 - 멀티캐스트 목적지 주소가 라우터 상태에 존재하지 않으면, 패킷을 절대적으로 전달하지 않는 것이 추천됨
 - 존재한다면 필터 모드 및 송신자 주소에 근거하여 권장 사항이 결정됨

Filter Mode	Source Address	Source Timer Value	Recommendation
Include	In the list	Timer > 0	Forward
Include	In the list	Timer = 0	Do not forward
Include	Not in the list	N/A	Do not forward
Exclude	In the list	Timer > 0	Do not forward
Exclude	In the list	Timer = 0	Forward
Exclude	Not in the list	N/A	Forward

IGMP

- 캡슐화

- IGMP 메시지는 IP 데이터그램에 캡슐화됨
 - 프로토콜 필드는 2, TTL 필드는 1로 설정됨
- 데이터그램의 목적지 IP 주소는 메시지 유형에 따라 변경됨

- 이전 버전과의 호환성

- 버전 1, 2와의 호환성을 위해 버전 1, 2에 정의된 메시지를 수신하도록 설계됨

Message Type	IP Address
General Query	224.0.0.1
Other Queries	Group address
Report	224.0.0.22

Version	Type Value	Message Type
1	0x11	Query
	0x12	Membership Report
2	0x11	Query
	0x16	Membership Report
	0x17	Leave Group

목 차

- 유니캐스트 라우팅 프로토콜

- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

- 멀티캐스트 라우팅 프로토콜

- 유니캐스팅과 멀티캐스팅
- IGMP (Internet Group Management Protocol)
- 멀티캐스트 라우팅

멀티캐스트 라우팅

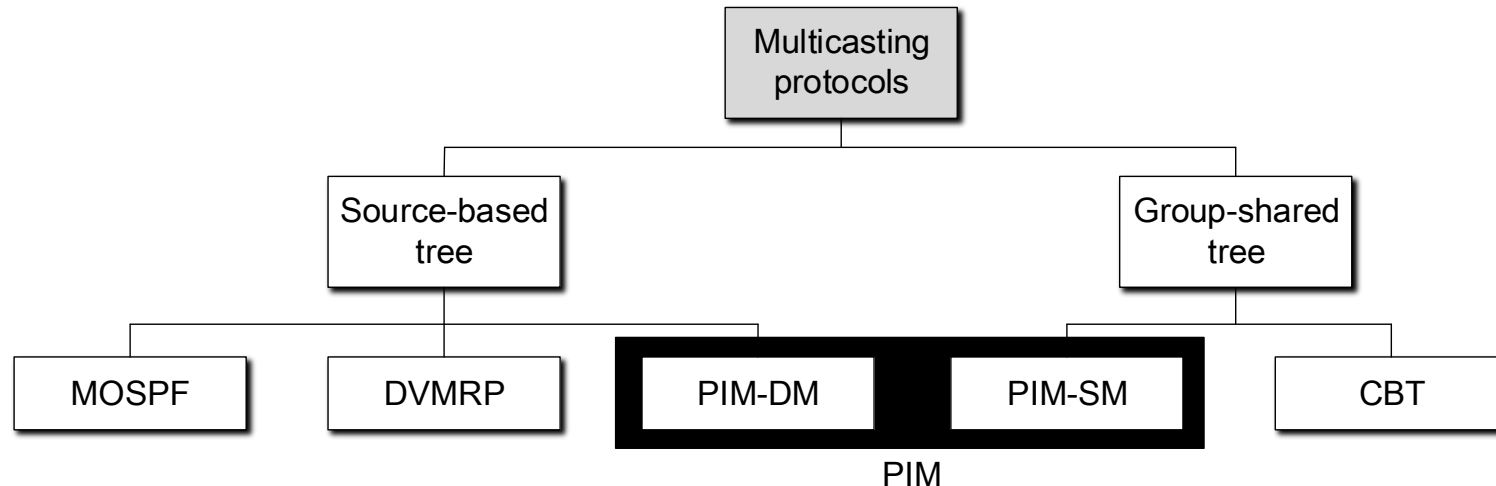
- 유형

- 송신자 기반 트리

- MOSPF (Multicast Open Shortest Path First)
- DVMRP (Distance Vector Multicast Routing Protocol)
- PIM-DM (Protocol Independent Multicast – Dense Mode)

- 그룹 공유 트리

- CBM (Core Based Tree)
- PIM-SM (PIM – Sparse Mode)



멀티캐스트 라우팅

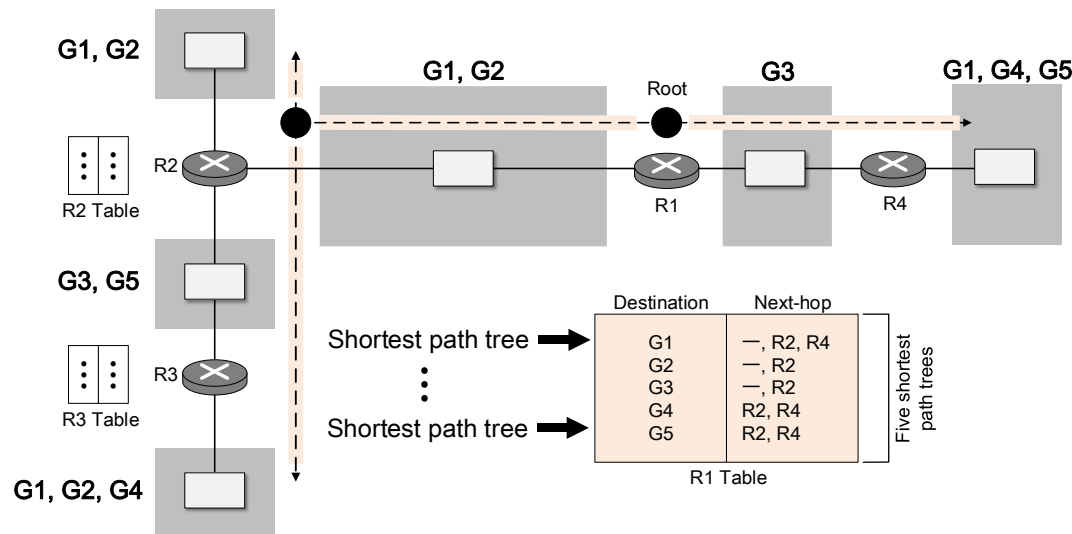
- 송신자 기반 트리 (1/8)

- 정의

- 멀티캐스트 송신자를 시작점으로 하여 각 수신자까지 최단 경로로 구성되는 트리

- 특징

- 각 라우터는 각 그룹 별로 하나의 최단 경로 트리를 가지는 것이 필요
- 그룹의 수가 증가할수록 라우터의 복잡도가 기하급수적으로 증가



멀티캐스트 라우팅

- 송신자 기반 트리 (2/8)

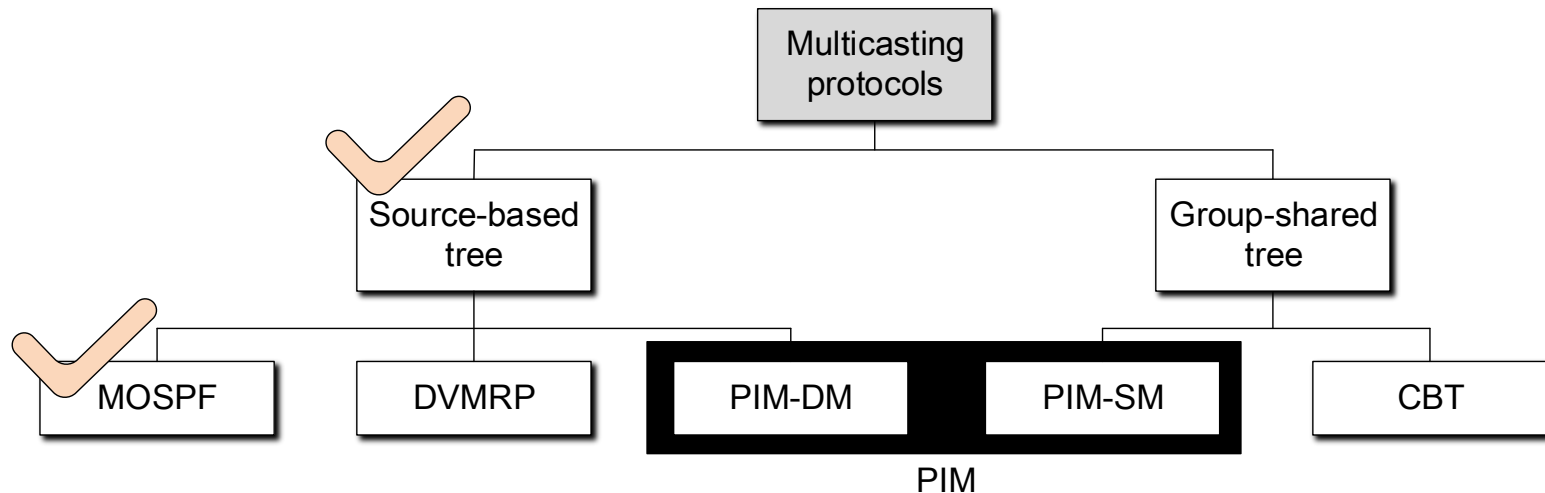
- 멀티캐스트 링크 상태 라우팅 (MOSPF) (1/2)

- 정의

- 송신자 기반의 트리를 생성하기 위해 멀티캐스트 링크 상태 라우팅을 사용하는 OSPF 프로토콜의 확장 프로토콜

- 특징

- 다익스트라 알고리즘으로 직접 최단 경로 트리 계산
 - 모든 라우터가 모든 그룹의 멤버십 정보를 알아야 하고, 활성화된 송신자마다 개별적으로 트리를 계산



멀티캐스트 라우팅

- 송신자 기반 트리 (3/8)

- 멀티캐스트 링크 상태 라우팅 (MOSPF) (2/2)

- 동작 과정

1. 각 노드는 자신의 링크 상에 멤버들이 존재하는 모든 그룹을 알림
 - IGMP를 운영하는 각 라우터들은 멤버 상태를 알기 위해 링크 상의 호스트들에게 그룹에 관한 정보를 얻음
2. 라우터가 모든 LSP (Link State Packet)를 수신하면 그룹의 수만큼의 토폴로지를 생성함
3. 다익스트라 알고리즘을 통해, 각 그룹에서 최단 경로 트리를 생성함

멀티캐스트 라우팅

- 송신자 기반 트리 (4/8)

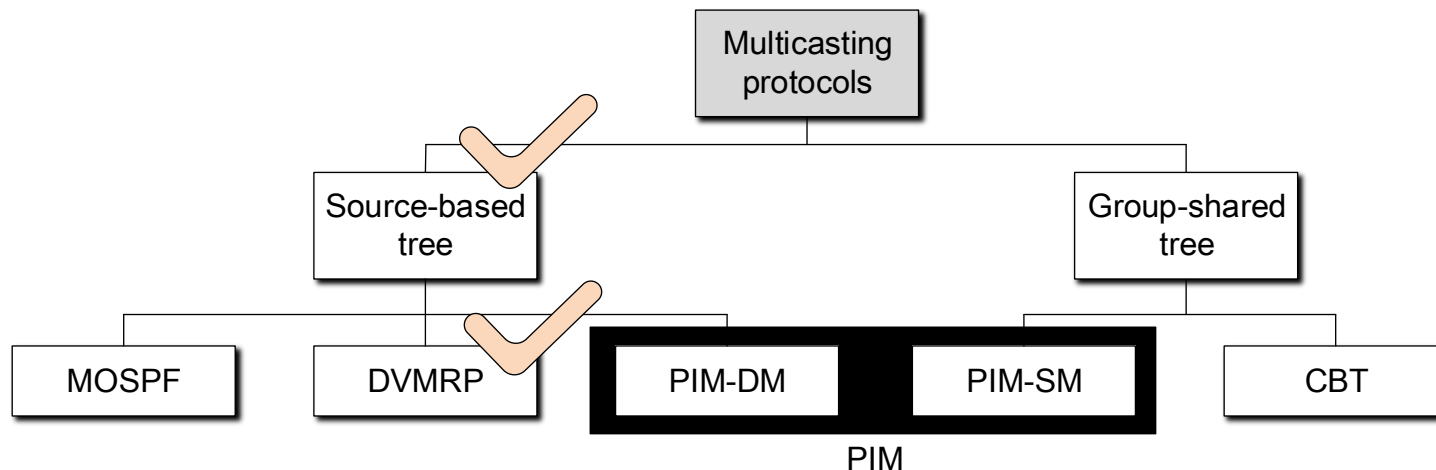
- 멀티캐스트 거리 벡터 라우팅 프로토콜(DVMRP) (1/5)

- 정의

- 거리 벡터 방식을 사용하여 멀티캐스트 트래픽을 라우팅하는 프로토콜

- 특징

- 송신자 기반 트리를 사용하나 라우터는 테이블을 만들지 않음
 - 4개의 판단-결정 정책에 기반을 둔 절차를 사용함



멀티캐스트 라우팅

- 송신자 기반 트리 (5/8)
- 멀티캐스트 거리 벡터 라우팅 프로토콜(DVMRP) (2/5)
 - 판단-결정 정책
 - 플러딩(Flooding)
 - 정의
 - 패킷을 수신하면 목적지 그룹 주소를 확인하지 않고 수신한 인터페이스를 제외한 모든 인터페이스로 내보내는 정책
 - 특징
 - 멀티캐스팅의 목적인 활성화 상태의 멤버를 가지는 모든 망이 패킷을 수신해야 한다는 목적은 달성함
 - 멀티캐스팅이 아닌 브로드캐스팅이라 라우터를 떠난 패킷이 다시 동일 라우터로 돌아오는 루프 문제가 발생할 수 있음

멀티캐스트 라우팅

- 송신자 기반 트리 (6/8)

- 멀티캐스트 거리 벡터 라우팅 프로토콜(DVMRP) (3/5)

- 판단-결정 정책

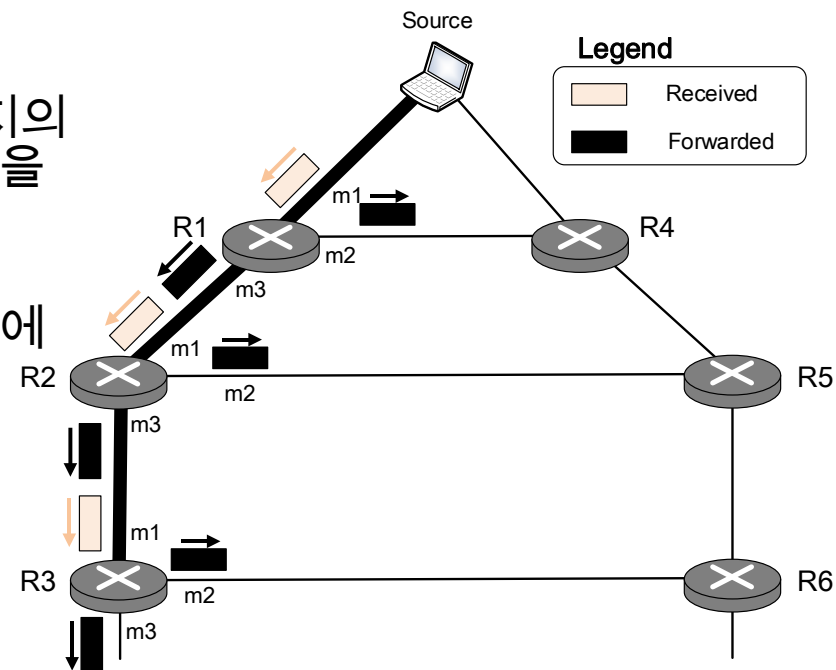
- 역 경로 전달(RPF, Reverse Path Forwarding)

- 정의

- 루프를 방지하기 위해 하나의 패킷만 전달하는 수정된 플러딩 정책

- 특징

- 라우터는 송신자로부터 라우터까지의 최단 경로를 통해 도착하는 사본만을 전달
 - 송신자로부터 해당 라우터까지 항상 하나의 최단 경로만 존재하기에 루프를 방지



멀티캐스트 라우팅

- 송신자 기반 트리 (7/8)

- 멀티캐스트 거리 벡터 라우팅 프로토콜(DVMRP) (4/5)

- 판단-결정 정책

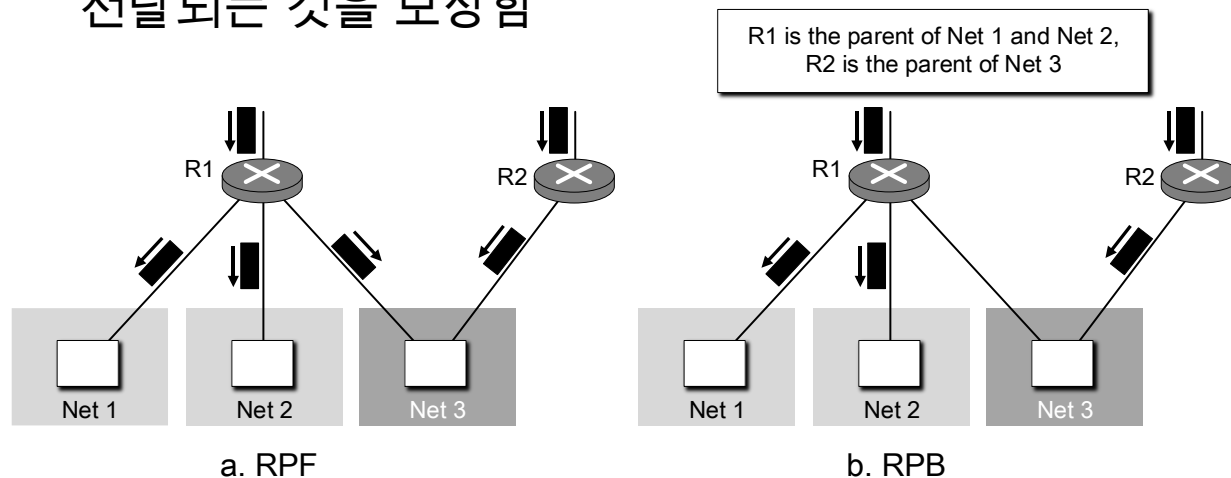
- 역 경로 브로드캐스팅(RPB, Reverse Path Broadcasting)

- 정의

- 각 송신자에 대해 자신이 지정된 부모 인터페이스로만 패킷을 전달하는 정책

- 특징

- 지정된 부모 라우터는 송신자로의 최단 경로를 가지는 라우터임
 - 패킷이 각 망에 도달하게 되는 것과 각 망에 반드시 하나의 사본만이 전달되는 것을 보장함



멀티캐스트 라우팅

- 송신자 기반 트리 (8/8)

- 멀티캐스트 거리 벡터 라우팅 프로토콜(DVMRP) (5/5)

- 판단-결정 정책

- 역 경로 멀티캐스팅(RPM, Reverse Path Multicasting)

- 정의

- 멀티캐스트 패킷이 특정 그룹에 대한 액티브 멤버를 가지는 망으로만 전달되는 정책

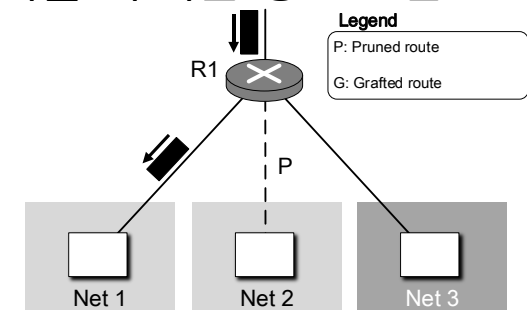
- 과정

- 제거

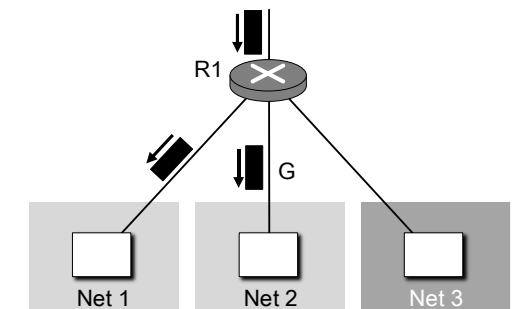
- 망에 연결된 라우터가 멀티캐스트 패킷을 수신하지 않고자 하면 제거 메시지 (Prune message)를 상위 라우터에 전송

- 접목

- 제거 메시지가 보내진 네트워크 중 한 곳에서 멀티캐스트 패킷을 수신하고자 할 때 접목 메시지 (Graft message)를 전송하여 상위 라우터가 멀티캐스트 패킷을 계속해 전송하도록 함



a. RPF(after pruning)



b. RPM(after grafting)

멀티캐스트 라우팅

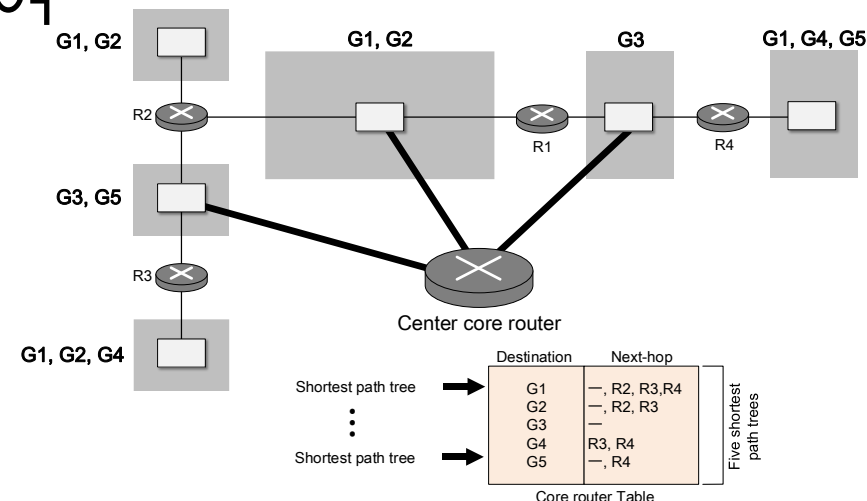
- 그룹 공유 트리 (1/4)

- 정의

- 미리 지정된 하나의 중심 라우터를 시작점으로 하여 그룹의 모든 멤버가 함께 사용하는 단일 트리

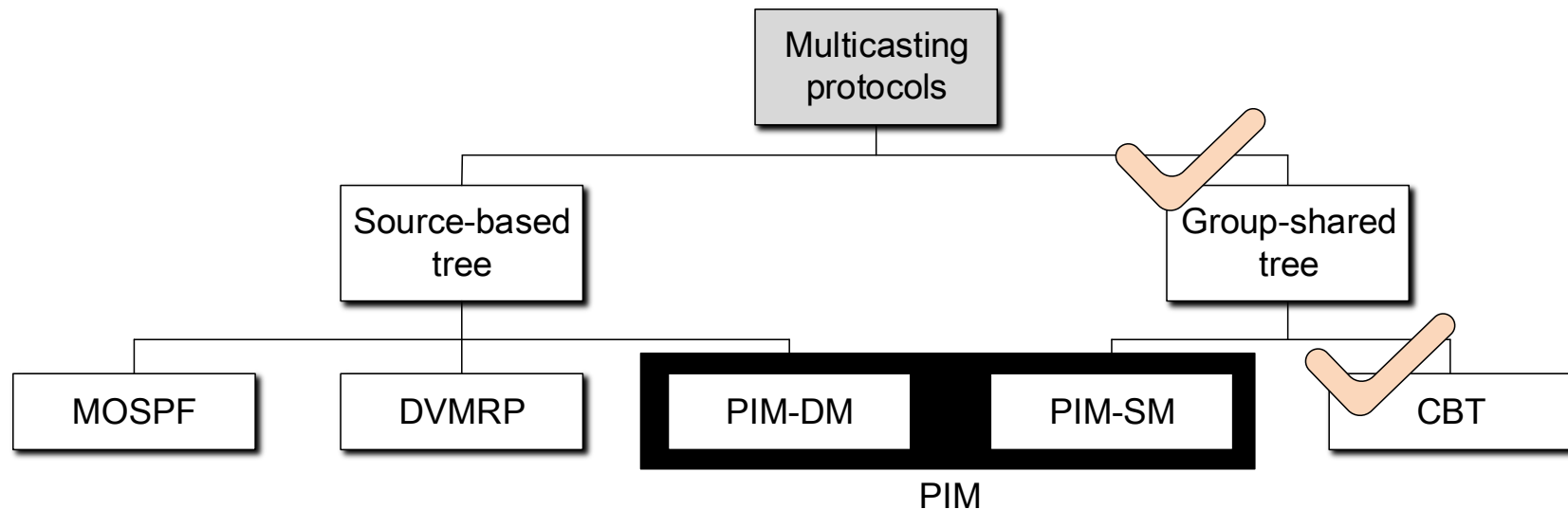
- 특징

- 중심 코어 라우터라고 불리는 하나의 지정된 랑데부 (Rendezvous) 라우터가 멀티캐스트 트래픽을 분배
- 코어 라우터만이 각 그룹에 해당하는 최단 경로 트리를 가지고 멀티캐스트 라우팅에 참여



멀티캐스트 라우팅

- 그룹 공유 트리 (2/4)
- 코어 기반 트리(CBT) (1/3)
 - 정의
 - 코어를 트리의 루트로 사용하는 공유 프로토콜
- 특징
 - AS는 지역으로 나뉘게 되고 각 지역마다 코어가 선택됨



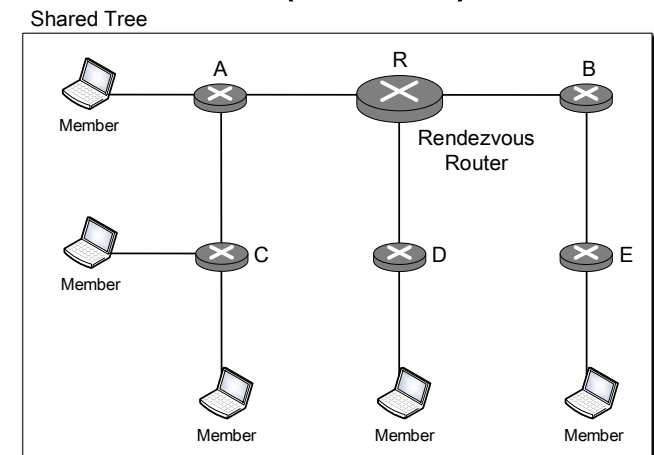
멀티캐스트 라우팅

- 그룹 공유 트리 (3/4)

- 코어 기반 트리(CBT) (2/3)

- 트리 형성

- 랑데부 포인트가 선택되면 모든 라우터는 랑데부 포인트의 유니캐스트 주소를 통보받음
 - 각 라우터는 그룹에 가입하고 싶음을 알리기 위해 유니캐스트 가입 (Join) 메시지를 전송함
 - 랑데부 라우터가 그룹의 모든 멤버들로부터 가입 메시지를 받게 되면 트리가 형성
 - 라우터가 그룹을 떠나고 싶으면 상위 라우터에 탈퇴(Leave) 메시지를 전송함



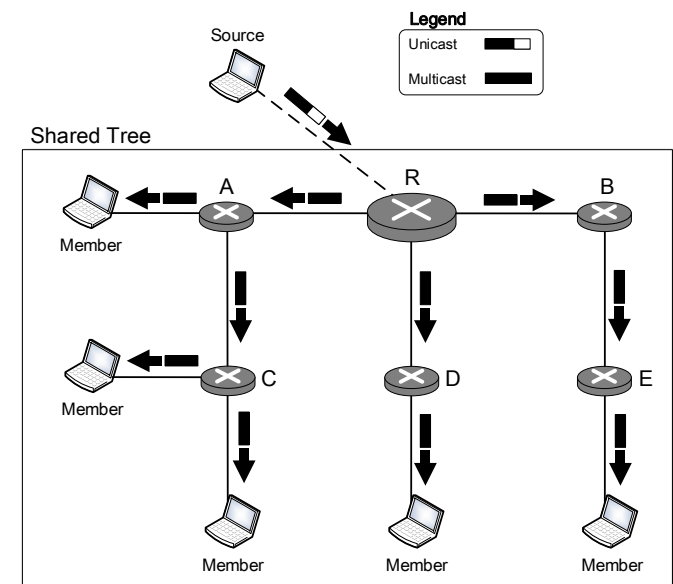
멀티캐스트 라우팅

- 그룹 공유 트리 (4/4)

- 코어 기반 트리(CBT) (3/3)

- 멀티캐스트 패킷 전송

1. 트리에 소속 유무에 상관 없이 송신자는 멀티캐스트 패킷을 코어의 유니캐스트 주소로 포장하여 코어로 전송
2. 코어 라우터는 유니캐스트 패킷의 헤더를 벗겨내고 남은 멀티캐스트 패킷을 수신 그룹에 인터페이스로 전달
3. 멀티캐스트 패킷을 수신한 각 라우터는 이어서 이를 모든 수신 그룹에 관한 인터페이스로 전달



멀티캐스트 라우팅

- 프로토콜 독립적 멀티캐스트(PIM, Protocol Independent Multicast) (1/5)

*역 경로 전달 (RPF, Reverse Path Forwarding)
루프를 방지하기 위해 하나의 패킷만 전달하도록
수정된 플러딩 정책

- 개요

- 이미 네트워크에 존재하던 다른 유니캐스트 프로토콜이 만들어 놓은 라우팅 테이블을 그대로 가져다 쓰는 프로토콜

- PIM-DM (PIM-Dense Mode)

- 멀티캐스트에 참여하는 밀집도가 높은 환경에서 사용
- 멀티캐스팅을 위해 *RPF와 제거/접속 정책을 사용하는 송신자 기반의 라우팅 프로토콜임

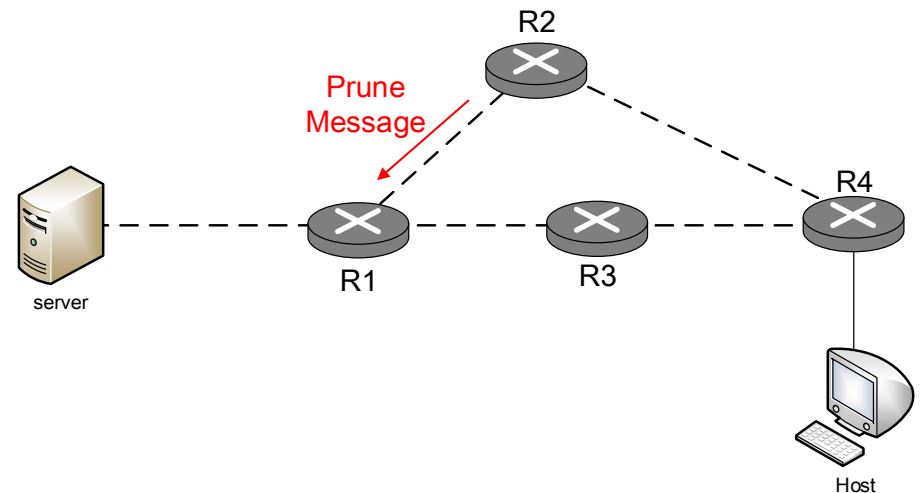
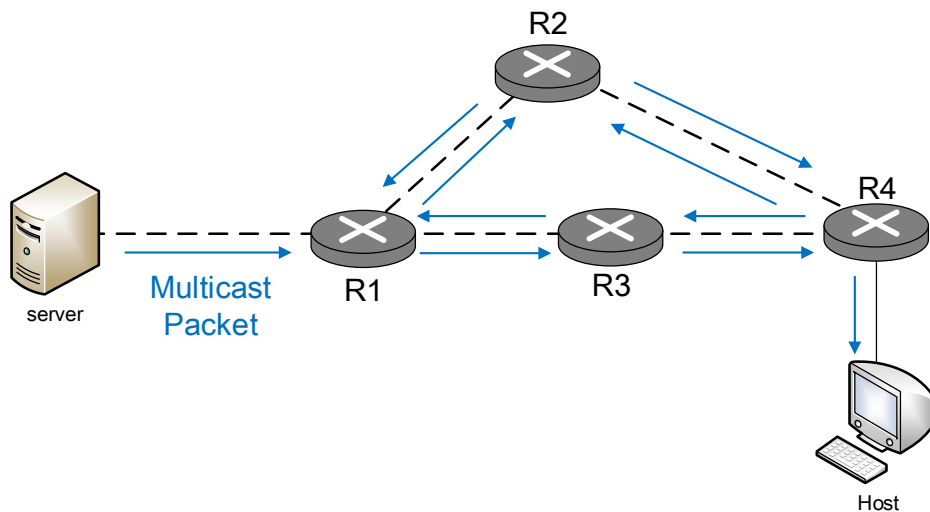
멀티캐스트 라우팅

- 프로토콜 독립적 멀티캐스트(PIM, Protocol Independent Multicast) (2/5)

- PIM-DM (PIM-Dense Mode)

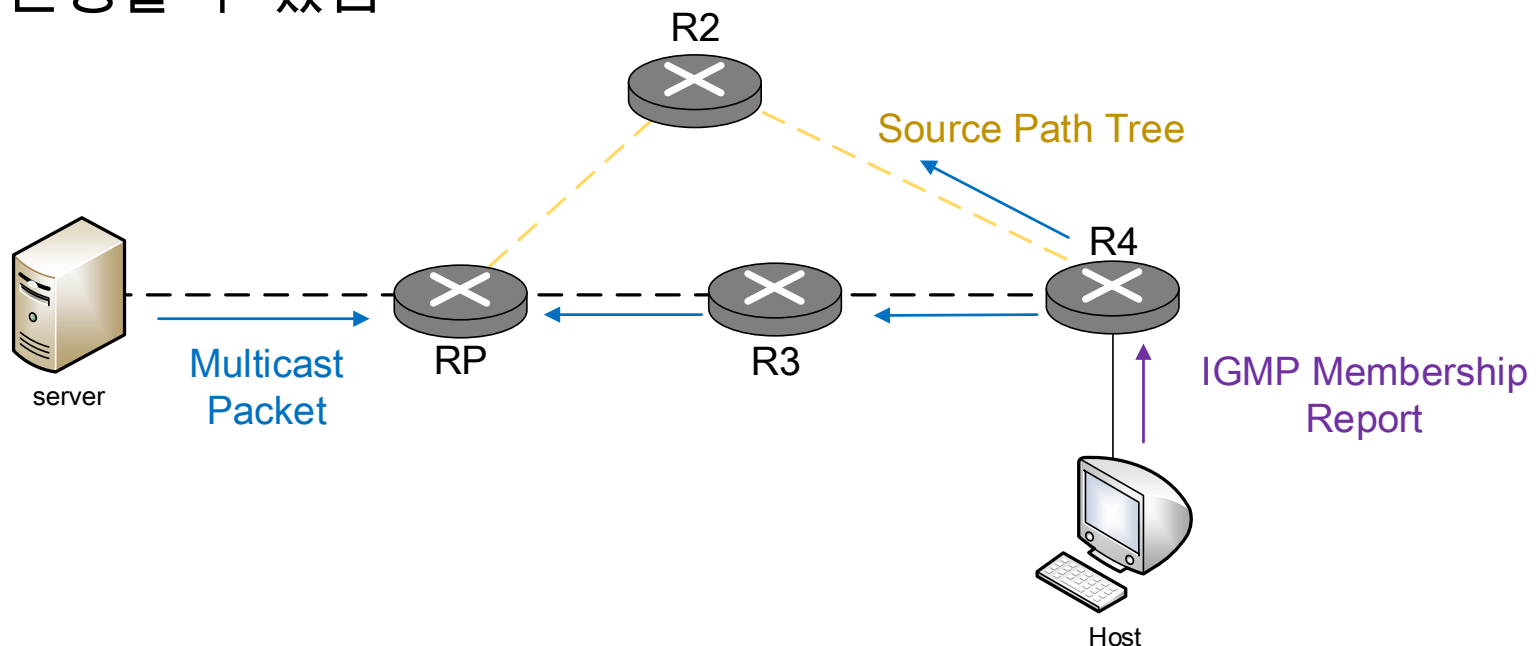
- 동작 방식

1. 멀티캐스트 송신자가 데이터를 보내기 시작하면, 첫 번째 라우터는 이 트래픽을 자신과 연결된 모든 곳에 플러딩
2. 데이터를 받은 라우터가 자신의 아래에 데이터를 받기 원하는 수신자가 없는 경우, 상위 라우터에게 데이터 전송 중지 요청



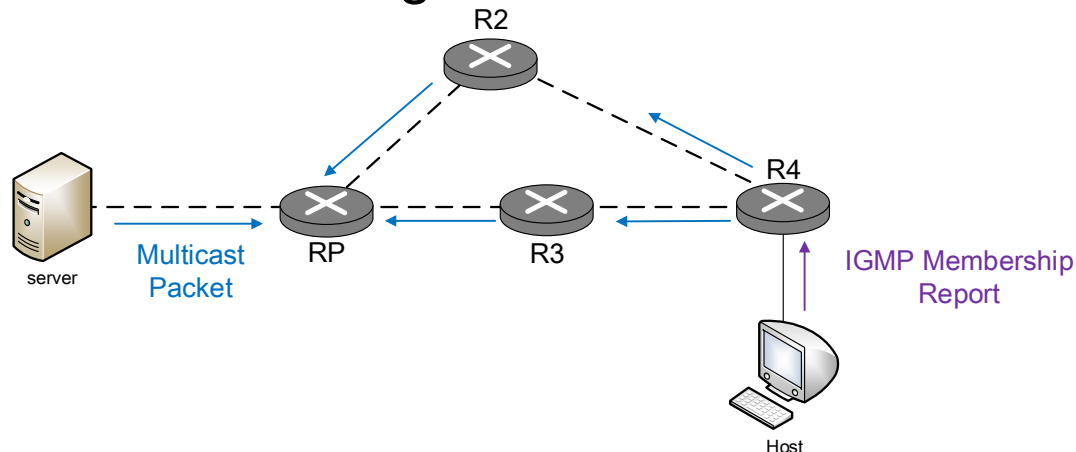
멀티캐스트 라우팅

- 프로토콜 독립적 멀티캐스트(PIM, Protocol Independent Multicast) (3/5)
- PIM-SM(PIM-Sparse Mode)
 - WAN과 같이 밀집도가 낮은 환경에 사용
 - 그룹 공유의 트리에서 필요시 송신자 기반의 트리로 변경
 - 라데부 포인트에서 멀리 떨어진 곳에 멀티캐스팅 영역이 있을 때 변경할 수 있음



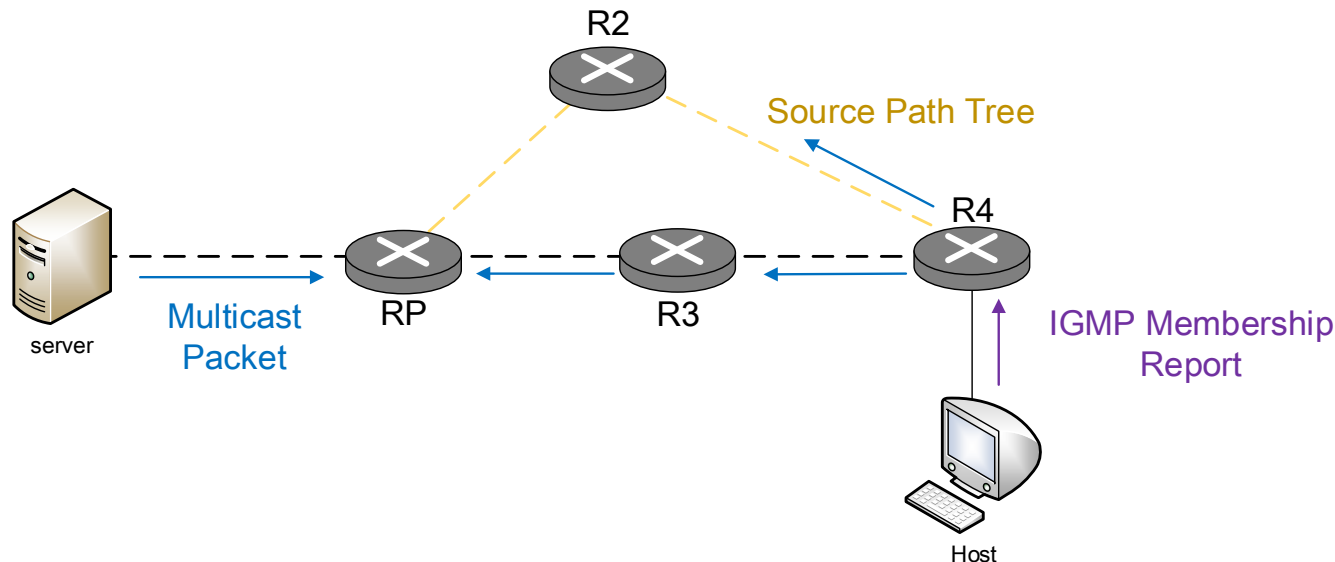
멀티캐스트 라우팅

- 프로토콜 독립적 멀티캐스트(PIM, Protocol Independent Multicast) (4/5)
- PIM-SM(PIM-Sparse Mode)
 - 동작 방식 (1/2)
 1. 랑데부 포인트(RP) 지정
 2. 특정 멀티캐스트 그룹의 데이터를 받고 싶은 수신자가 생기면, 수신자와 가장 가까운 라우터가 RP에게 메시지를 송신, 수신자와 RP를 잇는 경로가 생성
 3. 송신자가 데이터를 보내기 시작하면, 송신자와 가장 가까운 라우터가 RP에게 Register 메시지를 전송



멀티캐스트 라우팅

- 프로토콜 독립적 멀티캐스트(PIM, Protocol Independent Multicast) (5/5)
- PIM-SM(PIM-Sparse Mode)
 - 동작 방식 (2/2)
 4. RP는 등록된 데이터를 수신하고, 2번 과정에서 만들어 둔 경로를 따라 모든 수신자에게 데이터를 전달
 5. 데이터를 수신한 라우터는 RP를 거치지 않는 더 효율적인 경로를 파악하고, 해당 경로로 직접 연결을 요청



멀티캐스트 라우팅

- MBONE (Multicast Backbone)

- 정의

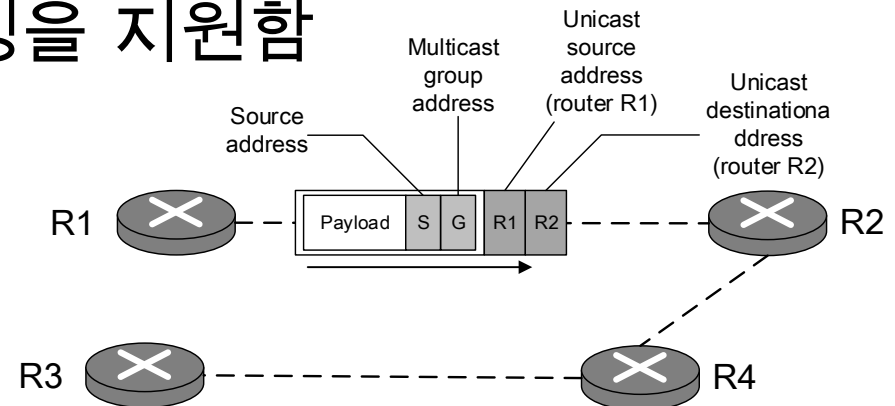
- 터널링을 통해 멀티캐스트 통신을 구현한 가상의 네트워크

- 동작 방식

- 멀티캐스트 패킷을 일반 유니캐스트 패킷 안에 캡슐화해 터널링 방식으로 전달

- 특징

- 최초의 대규모 멀티캐스트로 실시간 멀티캐스트 방송을 가능하게 함
- DVMRP에서 MBONE과 터널링을 지원함



Thanks!

민 윤 홍 (yunhong@pel.sejong.ac.kr)