

TCP/IP 프로토콜

-11장 유니캐스트 프로토콜-

서진우(jinwoo@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

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

목 차

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

보충

- 해결책 비교 (1/2)

- 포이즌 리버스(Poison Reverse)

- 정의

- 자신이 경유지로 사용하는 경로를 해당 경로를 학습한 이웃 라우터에게 메트릭 무한대로 광고하는 방식

- 동작 방식

- 1. A가 B를 통해 목적지 X로 가는 경로를 학습
 - 2. A는 B에게 X로의 경로를 무한대로 광고
 - 3. B는 A를 통해 X로 갈 수 없다고 판단하여 해당 경로를 선택하지 않음

- 효과

- 두 라우터가 서로를 다음 홉으로 선택하는 두 노드 라우팅 루프 방지
 - 수평 분할(Split Horizon)의 확장 방식으로 사용

보충

- 해결책 비교 (2/2)

- 라우트 포이즈닝(Route Poisoning)

- 정의

- 특정 목적지로의 경로가 실패하거나 유효하지 않은 경우, 해당 경로의 메트릭을 무한대로 설정하여 모든 이웃 라우터에게 즉시 광고하는 방식

- 동작 방식

- 1. 라우터에서 특정 목적지로의 경로에 장애 발생
 - 2. 해당 목적지의 메트릭을 무한대로 설정
 - 3. 변경된 경로 정보를 이웃 라우터에게 즉시 광고
 - 4. 이웃 라우터는 해당 경로를 사용할 수 없는 것으로 판단

- 효과

- 잘못된 라우팅 정보의 확산을 빠르게 차단하여 세 노드 이상의 라우팅 루프와 경로 불안정성 완화

목 차

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

도메인 내 및 도메인 간 라우팅

- 자율 시스템(AS, Autonomous System) (1/2)

- 정의

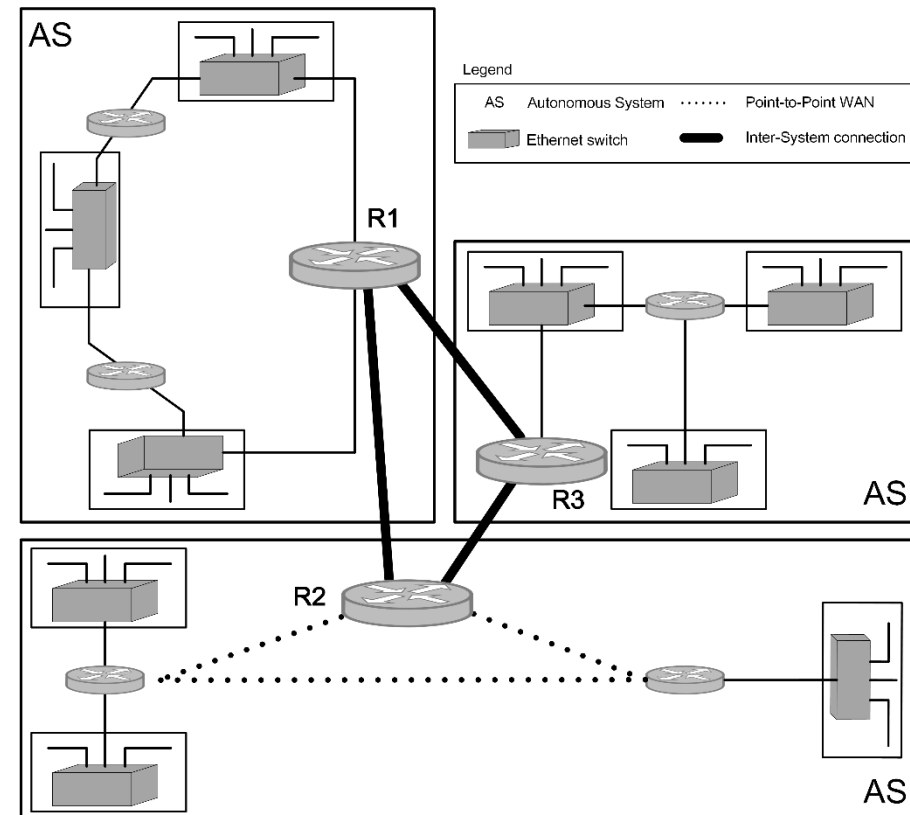
- 하나의 단일 관리 기간 하에 있는 라우터와 네트워크의 그룹
 - e.g., SKT, KT와 같은 ISP에서 제공하는 네트워크

- 도메인 내 라우팅(Intradomain Routing)

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

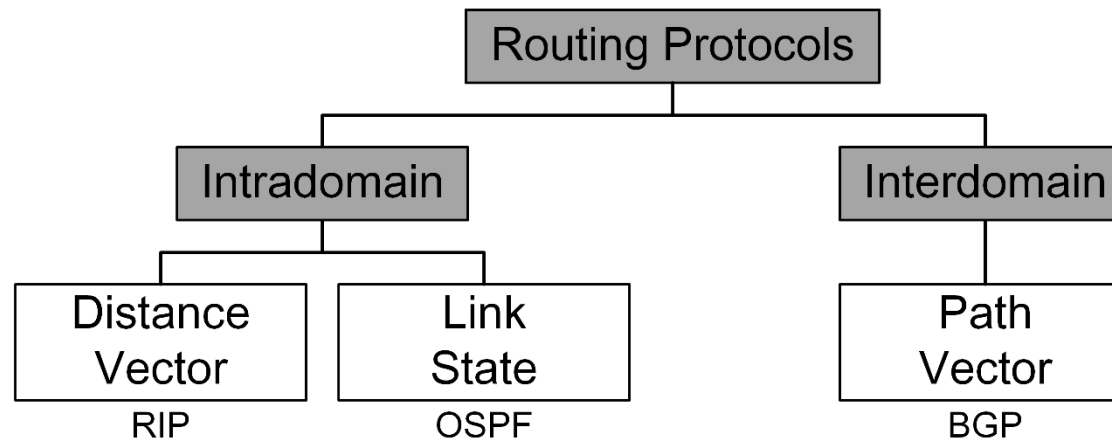
- 도메인 간 라우팅(Interdomain Routing)

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



도메인 내 및 도메인 간 라우팅

- 자율 시스템(AS, Autonomous System) (2/2)
- 라우팅 프로토콜
 - 도메인 내 라우팅 프로토콜
 - RIP (Routing Information Protocol)
 - 거리 벡터 라우팅
 - OSPF (Open Shortest Path First)
 - 링크 상태 라우팅
 - 도메인 간 라우팅 프로토콜
 - BGP (Border Gateway Protocol)
 - 경로 벡터 라우팅



목 차

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

거리 벡터 라우팅

- 거리 벡터 라우팅(Distance Vector Routing)
 - 정의
 - 네트워크 전체가 아닌, 인접한 라우터와의 거리 정보를 교환하여 경로를 설정하는 방식
 - 특징
 - 도메인 내부 라우팅에 유용
 - 도메인 간 라우팅에 확장된 개념 사용
 - 인접 라우터와 주기적으로 라우팅 테이블 정보 공유
 - RIP 경우 30초에 한 번 공유
 - Bellman-Ford 변형 알고리즘 사용
 - RIP (Routing Information Protocol)로 구현

거리 벡터 라우팅

- Bellman-Ford 알고리즘 (1/2)

- 정의

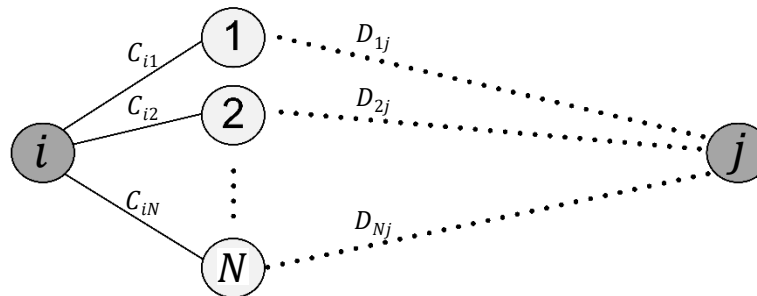
- 임의의 노드 간 비용을 통해 두 노드 간 최소 비용인 최소 거리를 구하는 알고리즘

- 특징

- 거리 벡터 라우팅의 경로 계산에 사용
- 각 노드 간 초기 정보가 모두 주어진 경우 사용
 - e.g., 도시 간 도로 지도 적용
- 모든 간선을 반복 확인하여 최단 경로를 단계적으로 결정

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

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 원리

- 각 노드를 경유하는 경로의 비용을 비교하여 최단 거리 갱신

1. 각 노드는 자신과의 최단 거리 및 비용을 0으로 초기화
2. 직접 연결된 노드까지의 거리를 링크 비용으로 설정하고, 연결되지 않은 노드는 무한대로 설정
3. 각 노드를 경유하여 기존 최단 거리보다 작은 경로가 있는지 확인하고 최단 거리 갱신
4. 최단 거리 벡터에 더 이상 변경이 없을 때까지 3번 과정 반복

```
Bellman_Ford(  
  // Initialization  
  for (i = 1 to N; for j = 1 to N)  
  {  
    if(i == j)           $D_{ij} = 0, C_{ij} = 0$   
    else                  $D_{ij} = \infty, C_{ij} = \text{cost between } i \text{ and } j$   
  }  
  // Updating  
  repeat  
  {  
    for(i = 1 to N; for j = 1 to N)  
    {  
       $D_{ij} \leftarrow \text{minimum} [(C_{i1}+D_{j1}), \dots, (C_{iN}+D_{jN})]$   
    } // end for  
  } until (there was no change in previous iteration)  
} // end Bellman-Ford
```

거리 벡터 라우팅

• 거리 벡터 라우팅 알고리즘 (1/4)

• 개요

• Bellman-Ford 변형 알고리즘

- AS 내 각 라우터들에 대한 라우팅 테이블 생성을 목적

• 동작 과정 (1/2)

• 각 라우터는 3가지 정보를 가지고 있어야 함

- 목적지 망, 비용, 다음 홉(Next-hop) 정보
- 라우팅 테이블을 $Table$, $Table$ 에서 i 번째 행을 $Table_i$, 행 i 번째에 해당하는 세 가지 정보들을 $Table_i.dest$, $Table_i.cost$, $Table_i.next$

1. 연결된 두 노드 간 비용(홉 수)은 1로 설정

2. 각 라우터는 Bellman-Ford 알고리즘의 일부 수행

*홉(Hop)

데이터 패킷이 출발지에서 목적지로 이동할 때
라우터나 게이트웨이 같은 네트워크 장비를 거쳐
가는 한 단위

Distance_Vector_Algorithm()

```
{  
    // At Startup  
    for ( $i = 1$  to  $N$ )           //  $N$  is number of ports  
    {  
         $Table_i.dest$  = address of the attached network  
         $Table_i.cost$  = 1  
         $Table_i.next$  = --  
        send a record  $R$  about each row to each neighbor  
    } // end for loop
```

거리 벡터 라우팅

• 거리 벡터 라우팅 알고리즘 (2/4)

• 동작 과정 (2/2)

3. 각 라우터는 자신의 라우팅 테이블 갱신 후 그 결과를 인접 노드에게 전달하여 그들의 라우팅 테이블을 갱신할 수 있도록 함

- 이웃 노드로부터 받은 정보를 R (record)라 하며, 이는 $R.dest$ 와 $R.cost$

```
// Updating
repeat (forever)
{
    wait for arrival of a record R from a neighbor
    Update(R, T)    // Call update module
    for(i = 1 to N) // N is the current table size
        send a record R about each row to each neighbor
    } // end repeat
} // end Distance_Vector
```

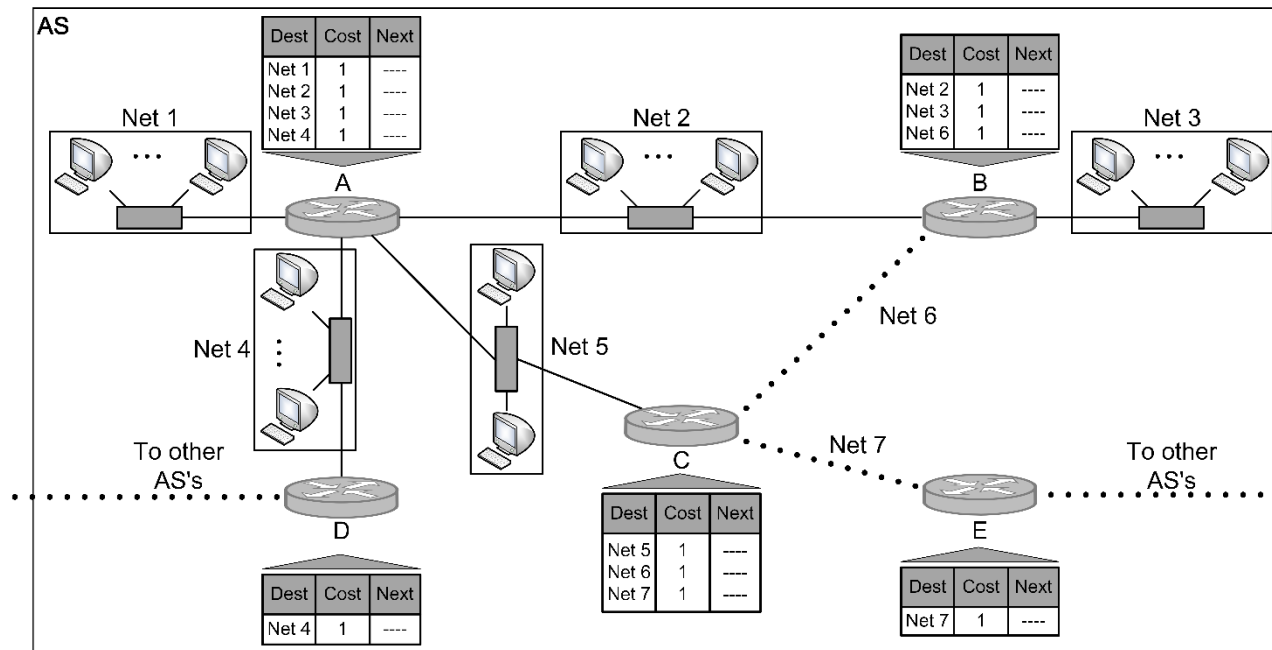
```
Update(R, T)
{
    search T for a destination matching the one in R
    if (destination is found in row i)
    {
        if ( $R.cost + 1 < T_i.cost$  or  $R.next == T_i.next$ )
        {
             $T_i.cost = R.cost + 1$ 
             $T_i.next = \text{Address of sending router}$ 
        }
        else
            discard the record // No change is needed
    }
    else
    {
         $T_{N+1}.dest = R.dest$ 
         $T_{N+1}.cost = R.cost$ 
         $T_{N+1}.next = \text{Address of sending router}$ 
        sort the table according to destination address
    }
} // end of Update module
```

거리 벡터 라우팅

• 거리 벡터 라우팅 알고리즘 (3/4)

• 예제 11.1 (1/2)

다음 그림은 AS에 대한 초기 라우팅 테이블을 보여준다. 그림은 모든 라우팅 테이블이 동시에 생성되는 것을 의미하지는 않는다. 각 라우터는 부팅될 때 자신만의 라우팅 테이블을 생성하게 된다. 이때 라우터 A가 이웃 노드인 라우터 B에 Net 1, Net 2, Net 4, Net 5 항목을 전송할 때, 라우터 B에서 라우팅 테이블 변화를 보이시오.

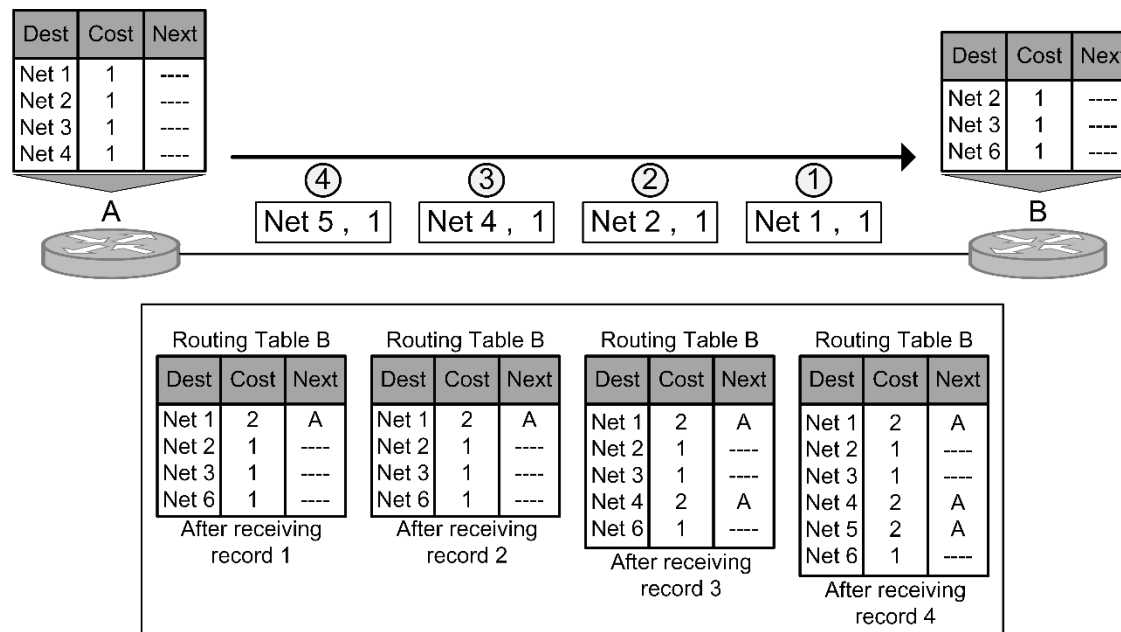


거리 벡터 라우팅

• 거리 벡터 라우팅 알고리즘 (4/4)

• 예제 11.1 (2/2)

다음 그림은 AS에 대한 초기 라우팅 테이블을 보여준다. 그림은 모든 라우팅 테이블이 동시에 생성되는 것을 의미하지는 않는다. 각 라우터는 부팅될 때 자신만의 라우팅 테이블을 생성하게 된다. 이때 라우터 A가 이웃 노드인 라우터 B에 Net 1, Net 2, Net 4, Net 5 항목을 전송할 때, 라우터 B에서 라우팅 테이블 변화를 보이시오.



거리 벡터 라우팅

- 거리 벡터 라우팅 문제점 (1/5)

- 무한대로의 카운트

- 정의

- 링크 오류로 인해 비용이 무한대로 바뀌었을 때, 이를 모든 라우터들이 인식하는데 시간이 소요되는 문제

- 고장 난 링크에 대한 비용이 모든 라우터에서 무한대로 기록되기까지 몇 차례의 갱신이 수행되어야 함

- 발생 원인

- 거리 벡터 라우팅에서 비용 감소와 같은 좋은 소식은 빠르게 확산되나, 비용 증가와 같은 나쁜 소식은 천천히 확산되어 발생

- 값이 줄어드는 계산은 즉시 갱신되고 주변에 알리는 반면, 값이 증가하는 계산은 기존 정보를 유지하려는 경향이 있음

거리 벡터 라우팅

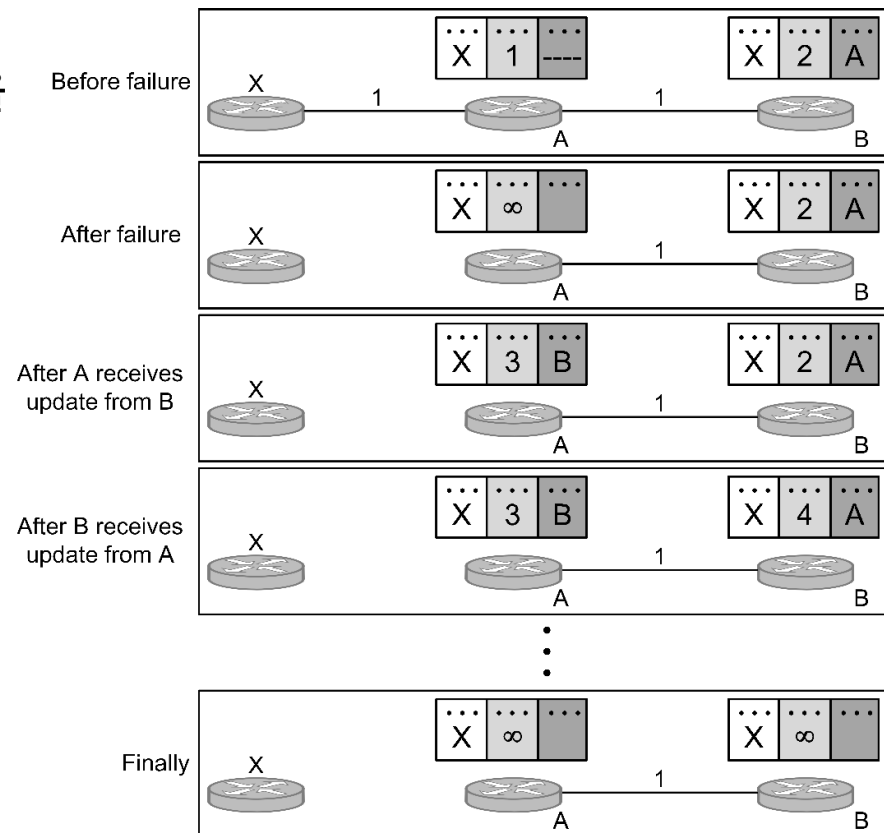
• 거리 벡터 라우팅 문제점 (2/5)

• 무한대로의 카운트

• 예시 - 두 노드 루프

• 발생 과정

1. A와 B는 노드 X에 도달하는 방법을 알고 있음
2. A와 X 사이의 링크가 실패하여, A는 자신의 테이블을 변경
3. B가 A의 라우팅 테이블 정보를 받기 전에, 자신의 라우팅 테이블 전송
4. A는 B의 라우팅 테이블 정보를 받고 자신의 라우팅 테이블을 갱신한 내용을 B에게 전송
5. 위 과정을 반복하여, 비용이 무한대가 되어야 X에 도달할 수 없음을 알게 됨



거리 벡터 라우팅

- 거리 벡터 라우팅 문제점 (3/5)

- 무한대로의 카운트

- 예시 - 두 노드 루프

- 해결책

- 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(Flooding) 하는 대신 각 인터페이스를 통해 자신 테이블의 일부만을 전송

- 포이즌 리버스(Poison Reverse)

- 이웃 라우터들에게 자신이 경유지로 사용되는 경로의 거리를 무한대로 알림
 - 수평 분할 정책과 조합해서 사용

거리 벡터 라우팅

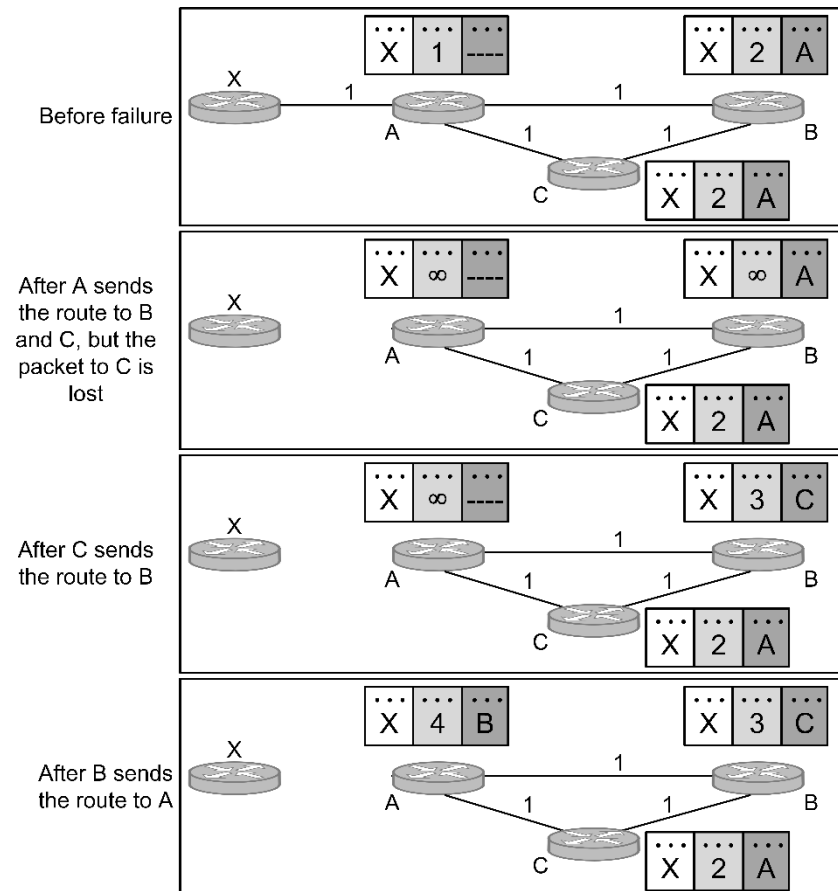
• 거리 벡터 라우팅 문제점 (4/5)

• 무한대로의 카운트

• 예시 - 세 노드 불안정성

• 발생 과정

1. A가 X에 도달 불가능할 것을 발견 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
4. B는 라우팅 테이블을 A에게 전달, A는 테이블을 갱신함
5. 위 과정을 반복하고, 각 노드의 비용이 무한대가 되면 중지됨



거리 벡터 라우팅

• 거리 벡터 라우팅 문제점 (5/5)

• 무한대로의 카운트

• 예시 - 세 노드 불안정성

- 두 노드 루프 해결 방법으로 해결이 보장되지 않아 새로운 해결 방법 필요

• 해결책

• 라우트 포이즈닝(Route Poisoning)

- 특정 목적지 경로가 실패하거나 유효하지 않은 경우, 그 목적지에 대해 메트릭을 무한대로 설정해 모든 이웃에게 즉시 광고하는 방식

*메트릭(Metric)

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

거리 벡터 라우팅

- 해결책 비교 (1/2)

- 포이즌 리버스(Poison Reverse)

- 정의

- 자신이 경유지로 사용하는 경로를 해당 경로를 학습한 이웃 라우터에게 메트릭 무한대로 광고하는 방식

- 동작 방식

- 1. A가 B를 통해 목적지 X로 가는 경로를 학습
 - 2. A는 B에게 X로의 경로를 무한대로 광고
 - 3. B는 A를 통해 X로 갈 수 없다고 판단하여 해당 경로를 선택하지 않음

- 효과

- 두 라우터가 서로를 다음 홉으로 선택하는 두 노드 라우팅 루프 방지
 - 수평 분할(Split Horizon)의 확장 방식으로 사용

거리 벡터 라우팅

- 해결책 비교 (2/2)

- 라우트 포이즈닝(Route Poisoning)

- 정의

- 특정 목적지로의 경로가 실패하거나 유효하지 않은 경우, 해당 경로의 메트릭을 무한대로 설정하여 모든 이웃 라우터에게 즉시 광고하는 방식

- 동작 방식

- 1. 라우터에서 특정 목적지로의 경로에 장애 발생
 - 2. 해당 목적지의 메트릭을 무한대로 설정
 - 3. 변경된 경로 정보를 이웃 라우터에게 즉시 광고
 - 4. 이웃 라우터는 해당 경로를 사용할 수 없는 것으로 판단

- 효과

- 잘못된 라우팅 정보의 확산을 빠르게 차단하여 세 노드 이상의 라우팅 루프와 경로 불안정성 완화

거리 벡터 라우팅

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

- 정의

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

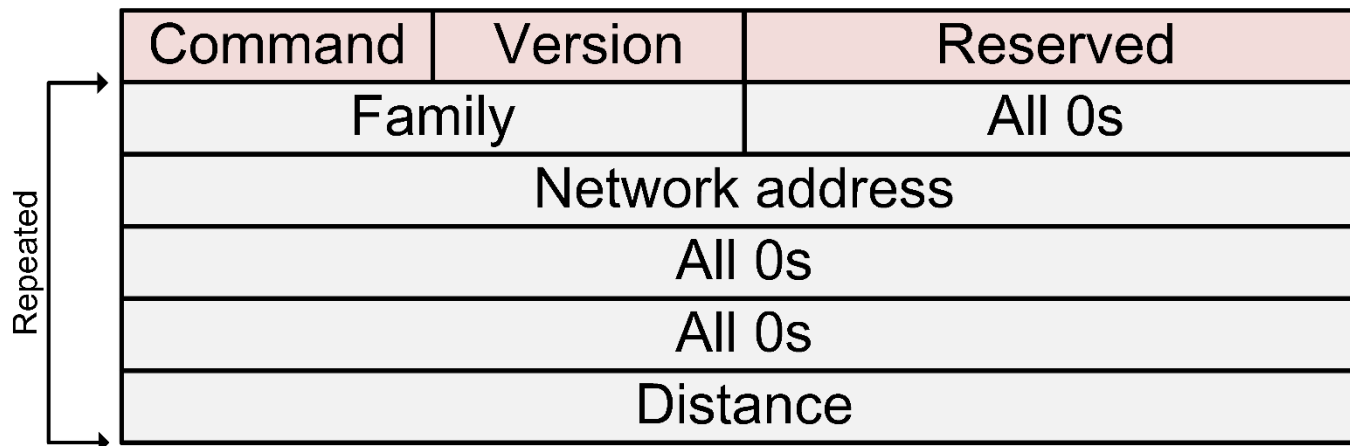
- 특징

- 자율 시스템 내부에서 사용되는 도메인 내 라우팅 프로토콜
- 거리 벡터 라우팅의 기반이 되는 프로토콜
- 홉 수를 메트릭으로 사용
- 무한대는 16으로 정의
 - 자율 시스템 내의 어떤 경로도 15홉 이상을 가지지 못 함을 의미
- 다음 홉 열은 패킷이 목적지에 도달하기 위해 보내져야 하는 라우터의 주소를 정의

거리 벡터 라우팅

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

필드	역할	크기
명령(Command)	메시지의 종류 (요청은 1, 응답은 2)	8비트
버전(Version)	RIP 버전	8비트
계열(Family)	사용된 프로토콜의 계열 (TCP/IP의 경우 2)	16비트
네트워크 주소(Network address)	목적지 네트워크 주소 (사용하고 남은 주소는 0)	14비트
거리(Distance)	송신 라우터로부터 목적지 네트워크까지의 홑 수	32비트



거리 벡터 라우팅

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

• 요청과 응답

• 요청

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

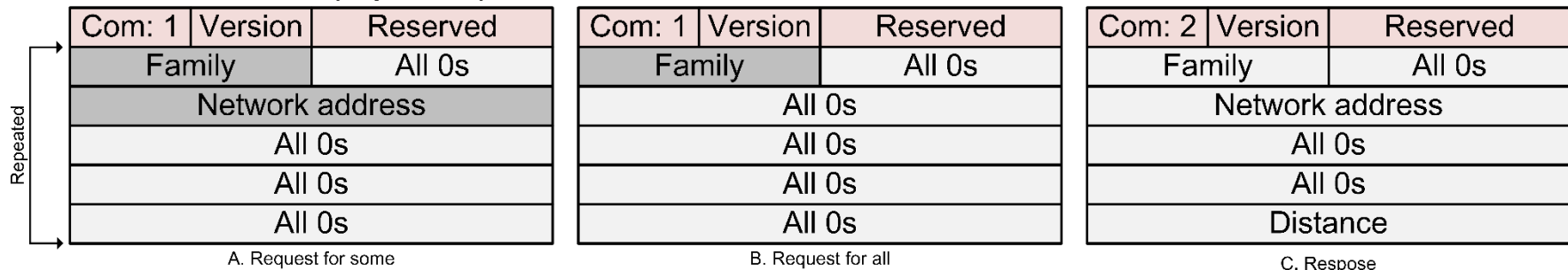
• 응답

• 요청된 응답

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

• 요청되지 않은 응답

- 30초 주기로 전송되며 모든 라우팅 테이블을 다루는 정보를 포함
- 갱신(Update)패킷이라고도 불림

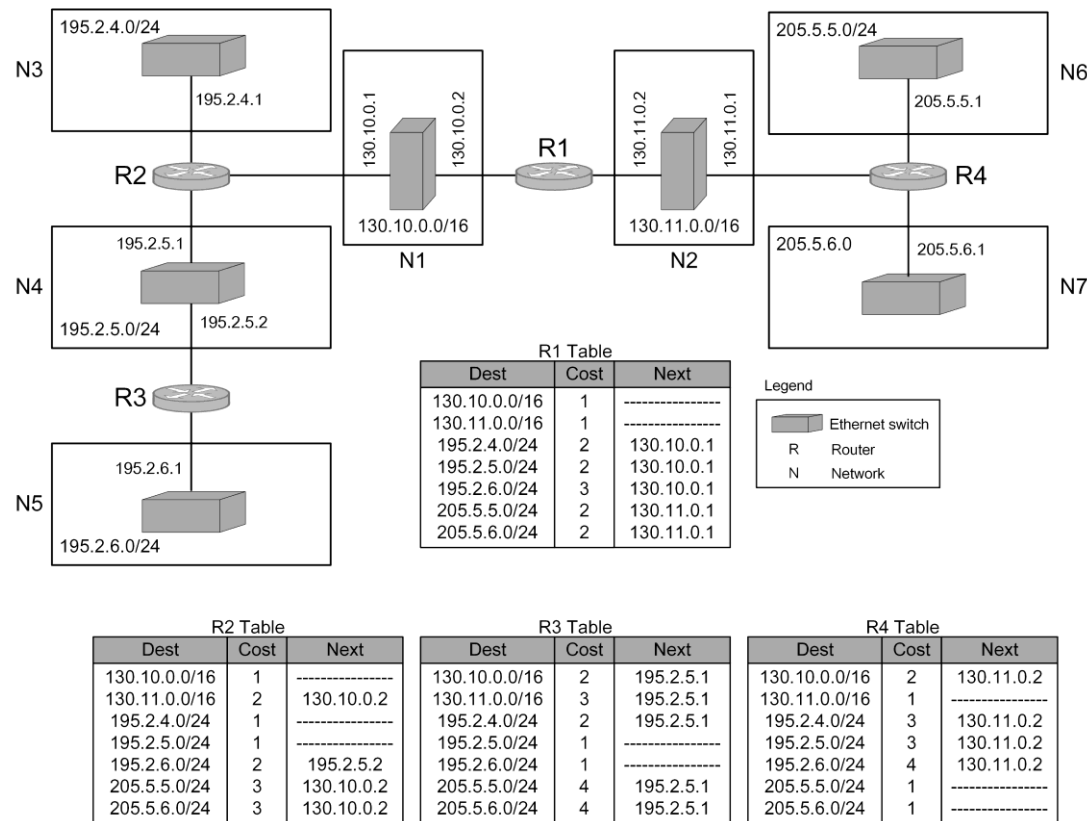


거리 벡터 라우팅

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

• 예제 11.2 (1/2)

아래 그림을 바탕으로, R1에서 R2로 보내지는 갱신 메시지는 보여라. 메시지는 130.10.0.2 인터페이스를 통해 보내진다.



거리 벡터 라우팅

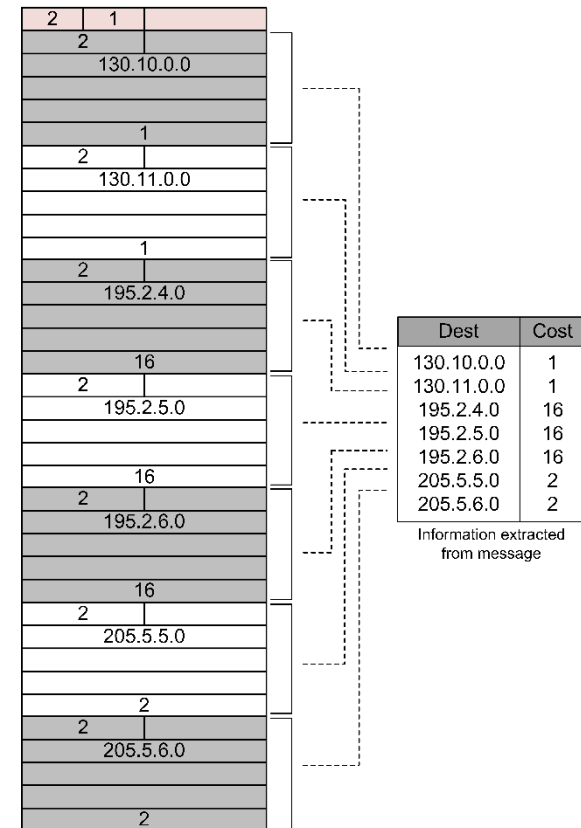
• RIP (Routing Information Protocol) (5/11)

• 예제 11.2 (2/2)

아래 그림을 바탕으로, R1에서 R2로 보내지는 갱신 메시지는 보여라. 메시지는 130.10.0.2 인터페이스를 통해 보내진다.

• 풀이

- 수평 분할과 포이즌 리버스정책을 염두에 두고 R1은 195.2.4.0, 195.2.6.0 정보를 R2로 얻었음에 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)
 - 타이머 (1/2)
 - 목적
 - 잘못된 정보로 인한 문제를 방지하기 위해 사용
 - 종류 (1/2)
 - 주기적 타이머(Periodic Timer)
 - 정규 갱신 메시지의 전송 주기를 제어
 - 25~35초의 임의의 값을 사용
 - 타이머가 0이 되면 갱신 메시지를 전송하고 타이머를 다시 설정
 - 라우터들이 동시에 갱신 메시지를 전송할 때 발생할 수 있는 네트워크 과부하를 방지
 - 만료 타이머(Expiration Timer)
 - 경로의 유효성을 관리
 - 라우터가 해당 경로의 갱신 메시지를 수신하면 180초로 설정
 - 해당 경로의 새로운 갱신 메시지를 수신하면 타이머를 다시 설정
 - 180초 동안 갱신 메시지를 수신하지 못하면 해당 경로를 무효화하고 홉 수를 16으로 설정

거리 벡터 라우팅

- RIP (Routing Information Protocol) (7/11)
 - 타이머 (2/2)
 - 종류 (2/2)
 - 폐 경로 수집 타이머(Garbage Collection Timer)
 - 무효화된 경로 정보를 인접 라우터에 알리고 삭제하기 전까지 유지
 - 해당 경로에 대해 120초로 설정
 - 타이머가 0이 되면 해당 경로 정보를 라우팅 테이블에서 삭제

거리 벡터 라우팅

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

- 예제 11.3

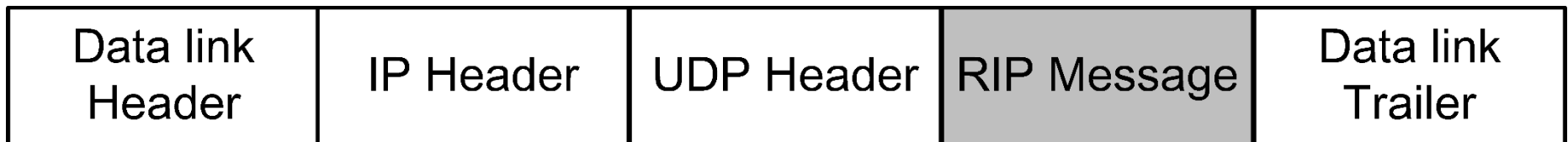
라우팅 테이블이 20개의 항목을 가지고 있다. 이 라우터가 200초 동안 5개의 경로에 대한 정보를 받지 못했다고 하면 이 시간에 가동되고 있는 타이머는 몇 개인가?

- 풀이
 - 주기적 타이머는 1개
 - 만료 타이머는 $20 - 5$ 이므로, 15개
 - 폐 경로 수집 타이머는 5개

∴ 21개

거리 벡터 라우팅

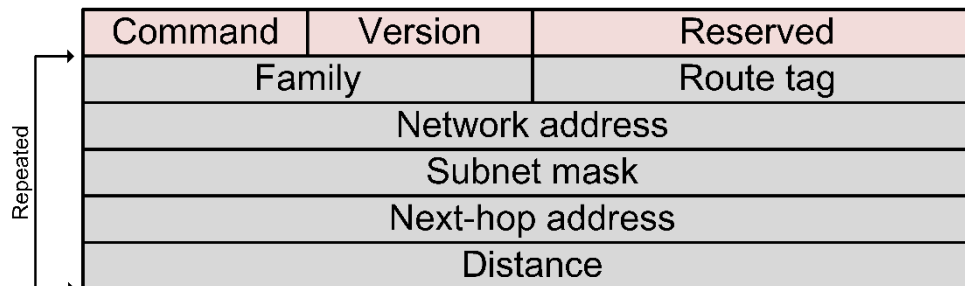
- RIP (Routing Information Protocol) (9/11)
- 캡슐화(Encapsulation)
 - RIPv2 메시지는 UDP 사용자 데이터그램에 캡슐화
 - RIPv2 메시지는 길이를 나타내는 필드를 가지고 있지 않음
 - UDP 헤더에는 전체 길이를 알려주는 Length 필드가 존재
- RIPv2에 할당된 잘 알려진 포트는 UDP 520번 포트



거리 벡터 라우팅

- RIP (Routing Information Protocol) (10/11)
- RIPv2 (1/2)
 - 기존의 RIP 형식 중 TCP/IP 프로토콜에서 모두 0으로 채워졌던 필드들을 새로운 필드로 대체한 방식
- 메시지 형식

필드	역할
경로 태그(Route tag)	자율 시스템의 번호화 같은 정보 전달 (도메인 간 라우팅 프로토콜의 정보의 수신을 가능하게 함)
서브넷 마스크(Subnet mask)	서브넷 마스크(또는 프리픽스)를 전달
다음 홉 주소(Next-hop address)	다음 홉의 주소를 알려줌 같은 네트워크를 공유할 경우 유용



거리 벡터 라우팅

- RIP (Routing Information Protocol) (11/11)
- RIPv2 (2/2)
 - RIP와의 차이점

RIP	RIPv2
클래스기반 주소 지정 (네트워크 당 호스트 수가 정해져 있기 때문에 주소 낭비 발생)	클래스없는 주소 지정 (고정된 클래스를 없애고 IP 주소를 필요한 만큼 할당하도록 설정)
-	인증되지 않은 광고에 대해 메시지를 보호하기 위해 인증 추가
모든 이웃에게 RIP 메시지를 보내기 위해 브로드캐스팅 사용 (네트워크에 있는 모든 라우터뿐만 아니라 호스트들도 패킷을 수신)	네트워크에 있는 라우터들만 RIP 메시지를 수신할 수 있도록 모든 라우터 멀티캐스트 주소(All-router Multicast Address)를 사용

Command	Version	Reserved
0xFFFF		Authentication type
Authentication data 16 bytes		
⋮		

목 차

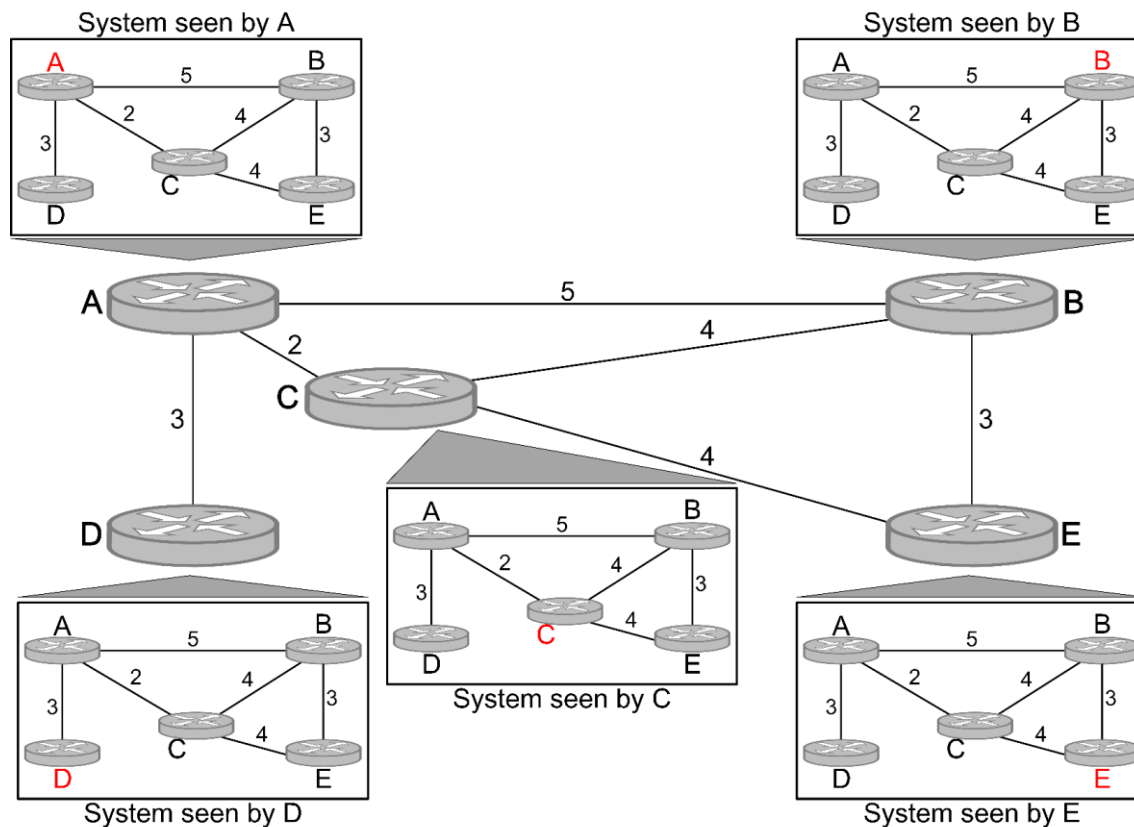
- 보충
- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

링크 상태 라우팅

- 링크 상태 라우팅(Link State Routing) (1/9)

- 정의

- 링크 상태 정보를 모든 라우터에게 전달해 최단 경로 트리를 구성하는 방식

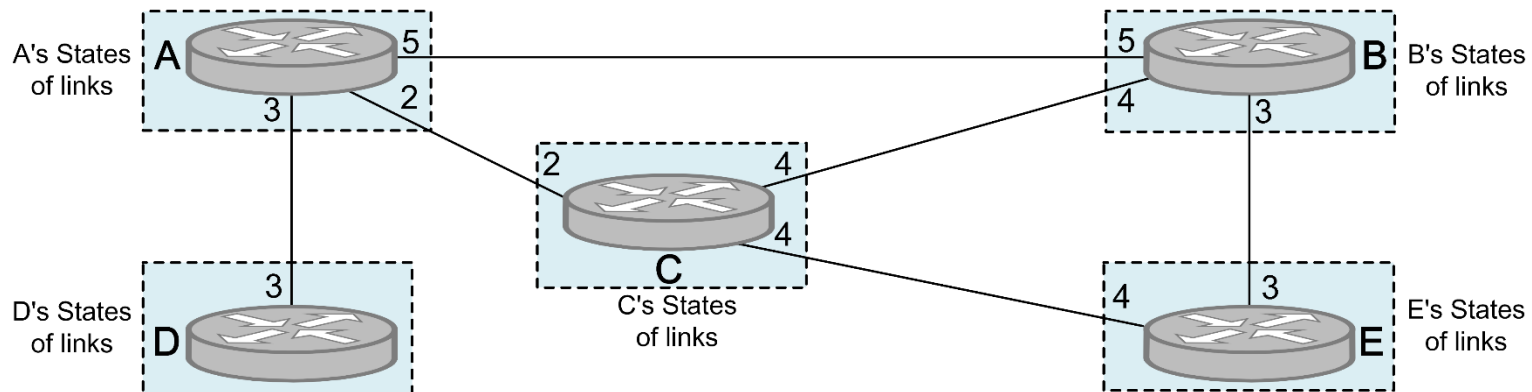


링크 상태 라우팅

• 링크 상태 라우팅(Link State Routing) (2/9)

• 특징

- 도메인 내부 라우팅에 유용
- 링크 상태 정보를 공유하며 최단 경로 계산
- 다익스트라(Dijkstra) 알고리즘 사용
- RIP보다 빠른 변화 감지 및 대응
 - 네트워크 장애 발생 시, 변화된 정보만 즉시 라우터에게 전파
- OSPF (Open Shortest Path First)로 구현



링크 상태 라우팅

- 링크 상태 라우팅(Link State Routing) (3/9)
- 다익스트라(Dijkstra) 알고리즘 (1/2)
 - 정의
 - 단일 출발 최단 경로(Single-Source Shortest Path) 문제를 해결하기 위한 그래프 탐색 알고리즘
 - 특징
 - 간선의 가중치가 항상 양수
 - 간선의 가중치가 음수일 경우, 최단 경로 계산 불가능
 - 최소 비용 간선 선택
 - 매 순간 가장 비용이 적은 간선을 선택
 - 동적 계획법
 - 이미 구한 최단 거리 정보를 작은 문제의 결과로 취급하여 이후의 계산에 사용

$$D_j = \min(D_j, D_i + C_{ij}) \quad \text{for all remaining nodes}$$

링크 상태 라우팅

• 링크 상태 라우팅(Link State Routing) (4/9)

• 다익스트라(Dijkstra) 알고리즘 (2/2)

• 동작 원리

1. 루트 노드 설정 후, 루트 노드의 모든 이웃에 대한 최단 경로를 루트와 그 이웃들 간의 비용으로 설정
 - 루트로의 최단 거리는 0으로 설정
2. 모든 노드들이 경로에 추가될 때 까지 반복 수행
 - 경로 상에 없는 노드를 검색하여 최단 거리를 가지는 것을 선택 후 경로에 추가
 - 2번째 절차에서 경로에 추가된 노드의 최단 거리를 사용하여 모든 나머지 노드들의 최단 거리 갱신

Dijkstra()

```
{
  // Initialization
  Path = {s}  // s means self
  for (i = 1 to N)
  {
    if (i is a neighbor of s and i ≠ s)   $D_i = C_{si}$ 
    if (i is not a neighbor of s)         $D_i = \infty$ 
  }
   $D_s = 0$ 
} // Dijkstra
// Iteration
Repeat
{
  // Finding the next node to be added
  Path = Path ∪ i  if  $D_i$  is minimum among all remaining nodes

  // Update the shortest distance for the rest
  for (j = 1 to M)  // M number of remaining nodes
  {
     $D_j = \text{minimum}(D_j, D_i + C_{ij})$ 
  }
} until (all nodes included in the path, M = 0)
```

링크 상태 라우팅

- 링크 상태 라우팅(Link State Routing) (5/9)
 - 라우팅 테이블 생성 과정 (1/5)
 1. 링크 상태 패킷(LSP, Link State Packet) 생성
 - 각 노드에 의해 링크 상태를 생성하는 것
 2. LSP 플러딩(Flooding)
 - 모든 다른 라우터에게 LSP들을 발송하는 것
 3. 다익스트라(Dijkstra) 알고리즘 기반의 최단 경로 트리 생성
 - 각 노드에서 최단 경로 트리 생성
 4. 라우팅 테이블 계산

링크 상태 라우팅

- 링크 상태 라우팅(Link State Routing) (6/9)
 - 라우팅 테이블 생성 과정 (2/5)
 1. 링크 상태 패킷(LSP, Link State Packet) 생성
 - 각 노드에 의해 링크 상태를 생성하는 것
 - 노드 식별자, 링크 리스트, 순서 번호 및 수명과 같은 최소량의 데이터만 전달한다고 가정
 - 도메인의 토폴로지에 변화가 있는 경우 생성
 - 도메인 내의 임의의 노드의 토폴로지를 갱신하도록 하기 위해 수행
 - 주기적으로 생성
 - 오래된 정보가 도메인에서 제거되도록 하기 위해 수행
 - 주기적 발송을 위한 타이머는 60분에서 2시간 범위
 - 더 긴 주기는 플러딩이 망에 너무 많은 트래픽을 유발하는 것을 방지

링크 상태 라우팅

- 링크 상태 라우팅(Link State Routing) (7/9)

- 라우팅 테이블 생성 과정 (3/5)

- 2. LSP 플러딩(Flooding)

- 모든 다른 라우터에게 LSP들을 발송하는 것
 - 노드는 LSP를 준비한 다음 이를 이웃만이 아닌 다른 모든 노드에게 발송해야함

1. LSP를 생성한 노드는 각 인터페이스를 통해 LSP의 복사본 전송
2. LSP를 수신한 노드는 이전 LSP와 비교
 1. 새 LSP가 오래되었다면, 해당 LSP 폐기
3. 새 LSP가 더 새 것이면, 다음 과정 수행
 1. 오래된 LSP 폐기 후 새로운 LSP 적용
 2. 복사본을 수신한 인터페이스를 제외한 다른 인터페이스로 전송하여 플러딩이 도메인 내의 어느 곳에서 중지되는 것을 방지

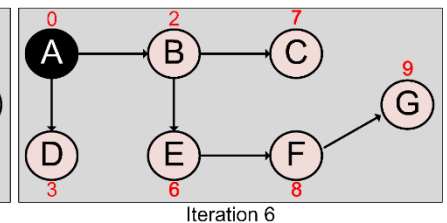
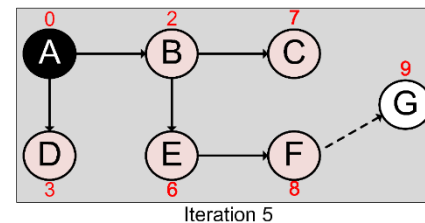
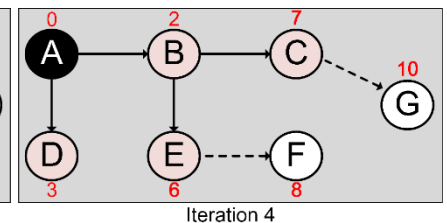
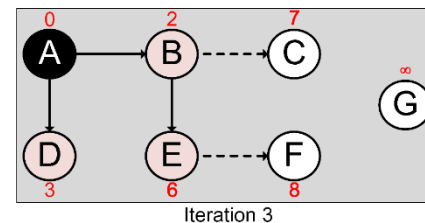
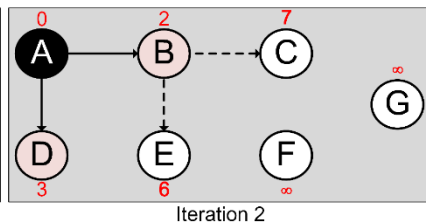
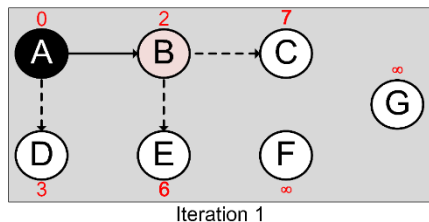
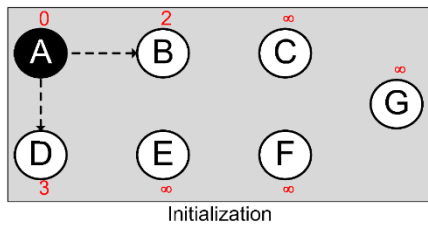
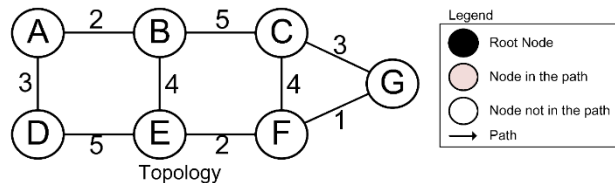
링크 상태 라우팅

• 링크 상태 라우팅(Link State Routing) (8/9)

• 라우팅 테이블 생성 과정 (4/5)

3. 다익스트라(Dijkstra) 알고리즘 기반 최단 경로 트리 생성

- 루트 노드 설정 후 인접한 노드들을 통해 최단 경로 트리 생성



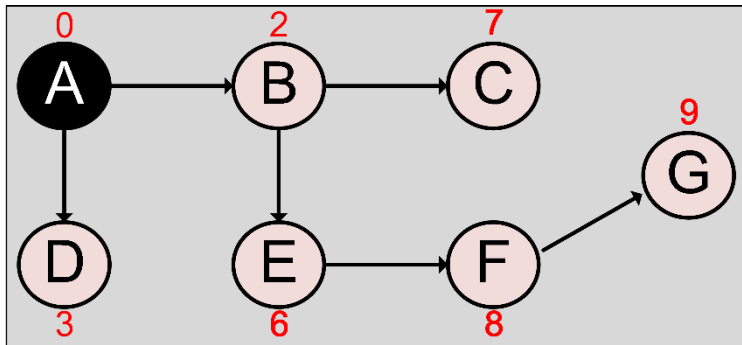
링크 상태 라우팅

• 링크 상태 라우팅(Link State Routing) (9/9)

• 라우팅 테이블 생성 과정 (5/5)

4. 라우팅 테이블 계산

- 각 노드는 다익스트라 알고리즘을 기반으로 한 최단 경로 트리를 사용하여 자신의 라우팅 테이블 구성
 - e.g., A의 라우팅 테이블



Iteration 6

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) (1/34)

• 정의

*지역(Areas)

자율 시스템 내에 포함되는 호스트, 라우터 및 네트워크의 모음

- 링크 상태 라우팅에 근거를 둔 도메인 내 라우팅 프로토콜

• 특징

- 링크 상태 라우팅 프로토콜
- 자율 시스템을 여러 지역(Areas)으로 분할
 - 대규모 네트워크에 적합
 - e.g., 기업의 대규모 사내 네트워크, 대학·기관 네트워크, ISP 내부 네트워크처럼 여러 라우터와 서브넷이 연결된 환경
- 메트릭(Metric)으로 비용(Cost)을 사용하여 최적 경로 선택
- 변경된 링크 상태 정보만을 전파하여 라우팅 정보 갱신

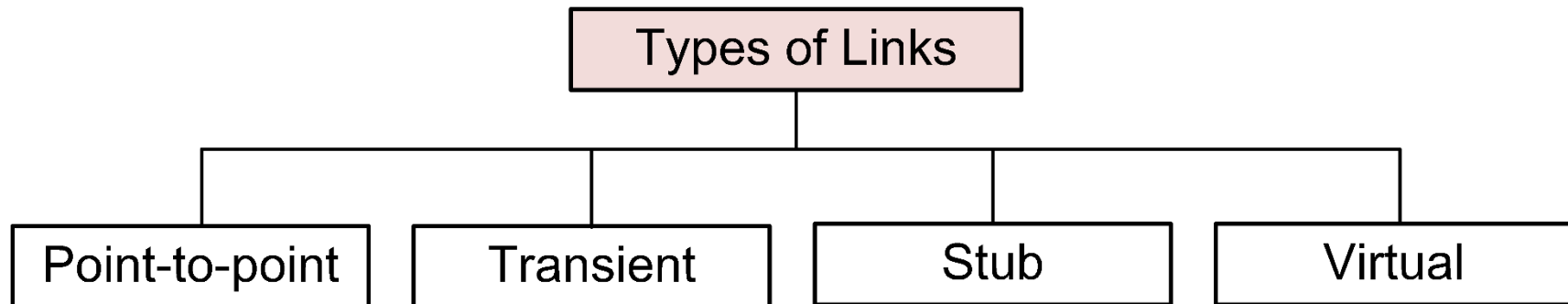
링크 상태 라우팅

• OSPF (Open Shortest Path First) (2/34)

• 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(Link)
OSPF 용어에서는 네트워크를 의미



링크 상태 라우팅

• OSPF (Open Shortest Path First) (3/34)

• 링크의 유형 (2/6)

• 점-대-점 링크(Point-to-point Link)

• 정의

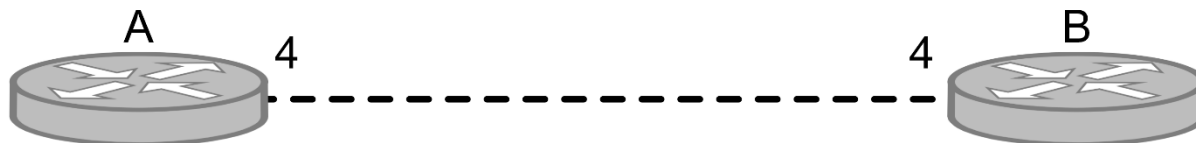
- 라우터 사이에 어떤 다른 호스트나 라우터 없이 직접 연결하는 링크

• 특징

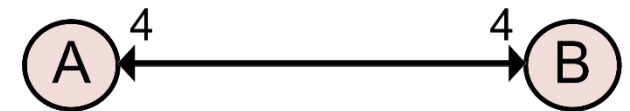
- 두 장치만 연결되므로 회선을 공유하는 다른 장치가 없음
- 단일 상대 라우터와 연결되므로 브로드캐스트나 다중 접근을 위한 네트워크 구성이 불필요
- 두 라우터를 노드로, 링크를 두 노드를 연결하는 양방향 링크로 표현

• 활용 예시

- 기업의 본사-지사 간 전용선
- 멀리 떨어져 있는 공장을 연결할 경우, 점-대-점 링크를 구축해 사용



A. Point-to-point network



B. Representation

링크 상태 라우팅

• OSPF (Open Shortest Path First) (4/34)

• 링크의 유형 (3/6)

• 경유 링크(Transient Link)

• 정의

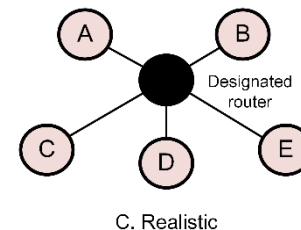
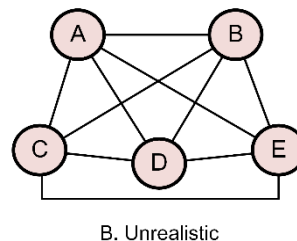
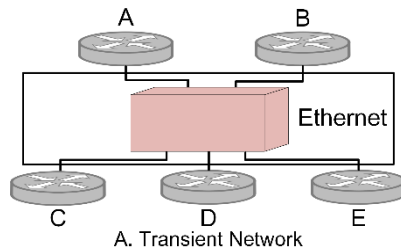
- 두 대 이상의 라우터가 함께 연결된 공유 네트워크를 나타내는 링크

• 특징

- 여러 라우터가 동일한 네트워크에 연결되어 데이터 교환
- 하나의 공유 네트워크에서 여러 라우터가 서로 통신 가능
- 모든 라우터가 일대일로 정보를 교환하면 라우터 수가 증가할수록 통신량 증가
 - 통신량을 줄이기 위해 지정 라우터(Designated Router, DR)를 사용

• 활용 예시

- Ethernet 스위치를 통해 여러 라우터가 동일한 LAN에 연결된 환경
- 여러 라우터가 하나의 공유 네트워크를 사용하는 기업 내부망



링크 상태 라우팅

- OSPF (Open Shortest Path First) (5/34)

- 링크의 유형 (4/6)

- 스템브 링크(Stub Link)

- 정의

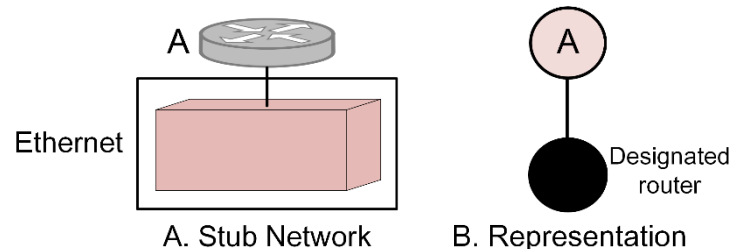
- 오직 하나의 라우터에만 연결된 네트워크

- 특징

- 네트워크 외부와의 통신은 연결된 하나의 라우터를 통해 통신
 - 다른 라우터와 직접 연결되지 않은 종단 네트워크를 의미
 - 라우터에서 해당 네트워크로 연결되는 형태로 표현
 - 하나의 연결 경로를 통해 외부 네트워크와 통신하므로 구조가 단순

- 활용 예시

- 일반 사용자 PC와 서버가 연결된 종단 LAN
 - 공장 설비나 센서 등이 연결된 사이버-물리 시스템(CPS, Cyber Physical System)



링크 상태 라우팅

- OSPF (Open Shortest Path First) (6/34)

- 링크의 유형 (5/6)

- 가상 링크(Virtual Link)

- 정의

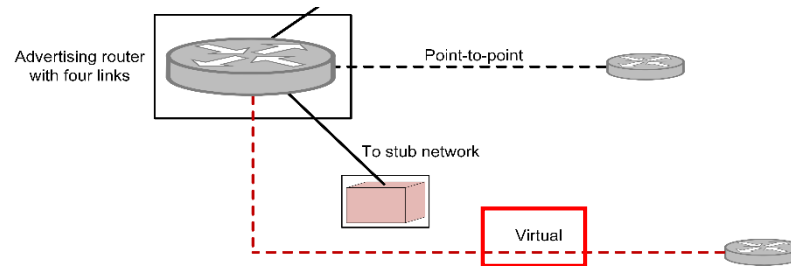
- 물리적으로 직접 연결되지 않은 라우터 또는 영역 사이를 논리적으로 연결하는 링크

- 특징

- 실제 물리 회선을 구축하지 않고 기존 네트워크를 통해 논리적인 연결 구성
 - 여러 라우터를 거쳐 두 지점을 연결하는 가상의 경로로 동작
 - 물리적인 연결이 끊어지거나 직접 연결할 수 없는 경우에도 활용

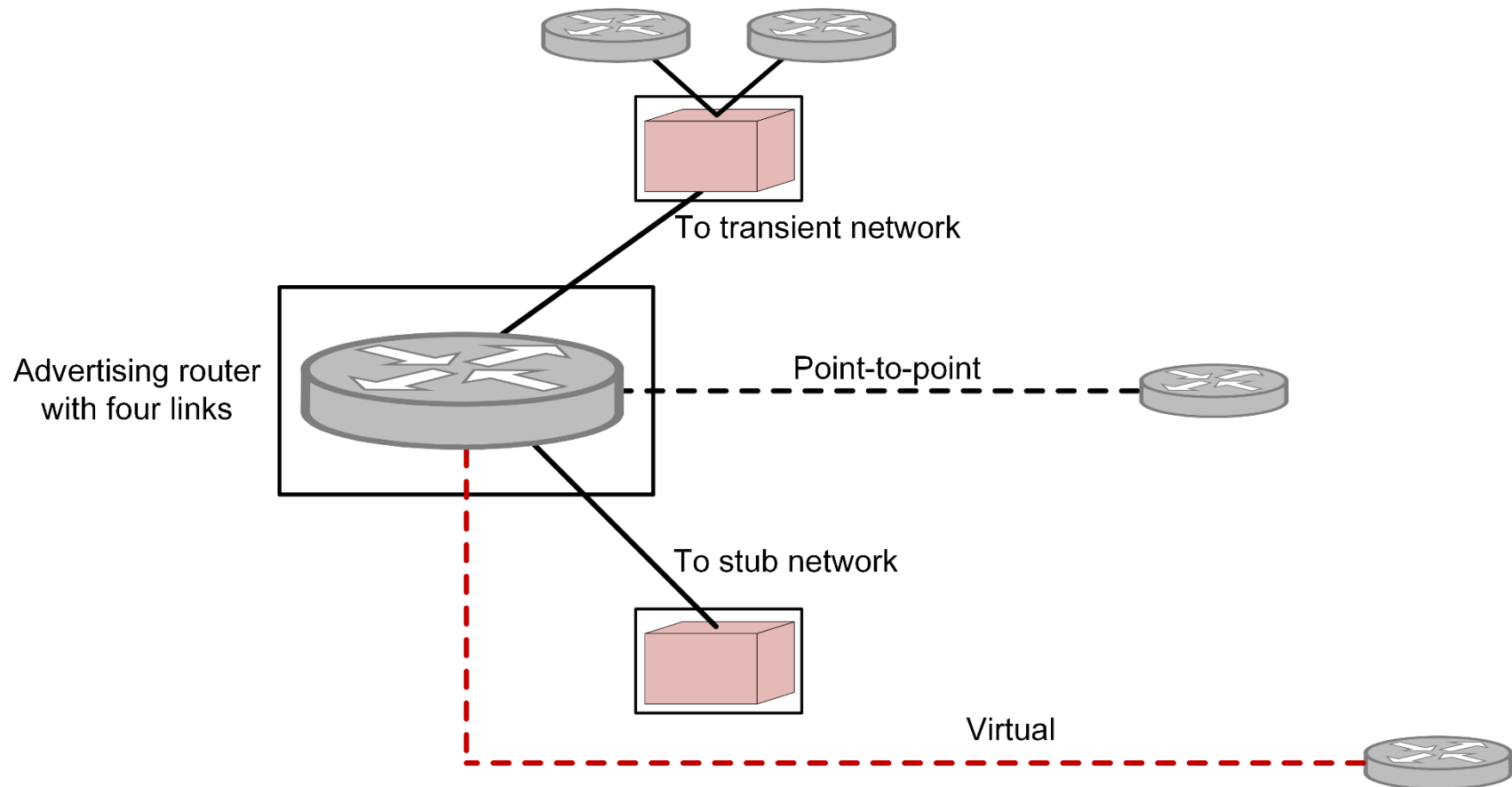
- 활용 예시

- 물리적으로 연결되지 않은 OSPF 영역을 연결하는 경우
 - 기존 네트워크 구조 변경이 어려운 상황에서 논리적인 연결을 구성하는 경우



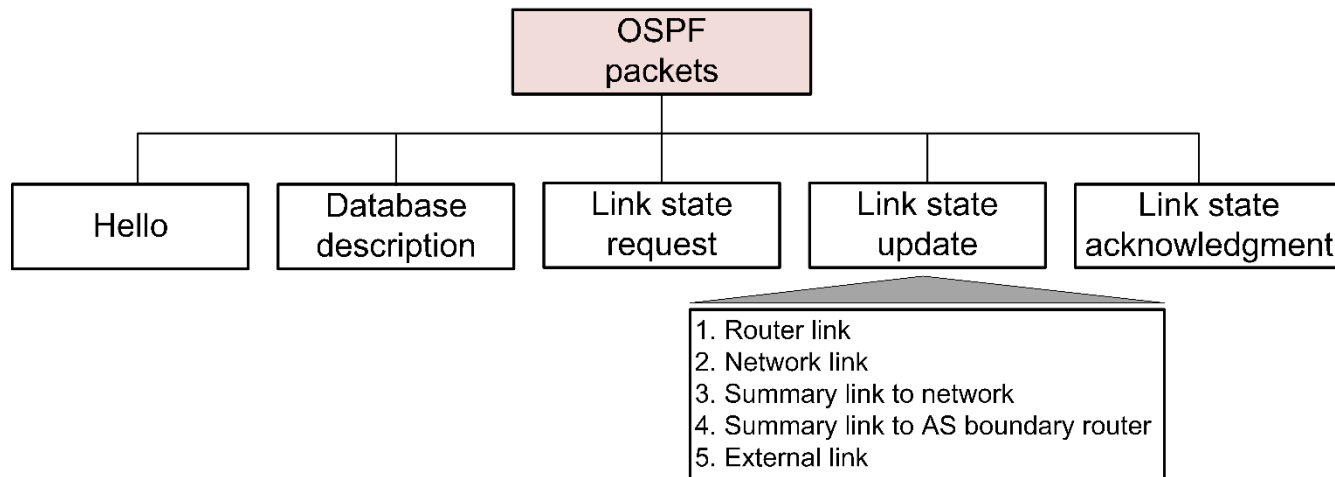
링크 상태 라우팅

- OSPF (Open Shortest Path First) (7/34)
- 링크의 유형 (6/6)
 - 4 가지 링크 유형 모식도



링크 상태 라우팅

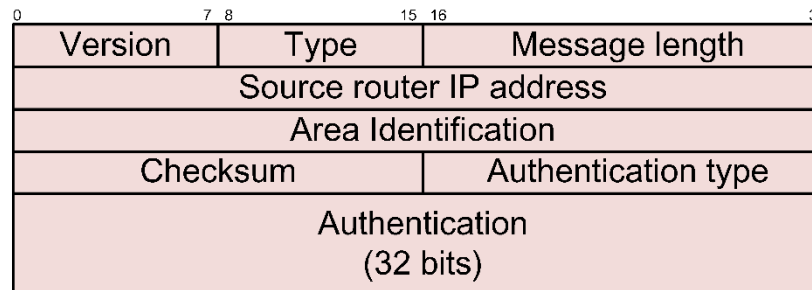
- OSPF (Open Shortest Path First) (8/34)
- OSPF 패킷 (1/22)
 - Hello 패킷
 - 데이터베이스 기술(Description) 패킷
 - 링크 상태 요청 패킷
 - 링크 상태 갱신 패킷
 - 링크 상태 확인응답 패킷



링크 상태 라우팅

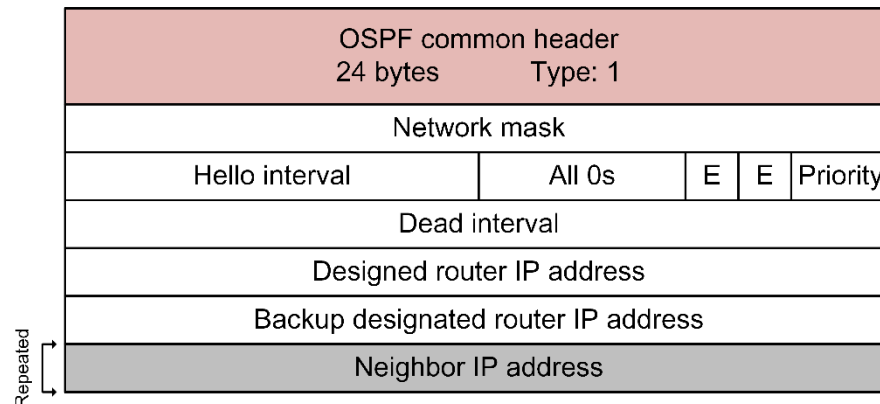
- OSPF (Open Shortest Path First) (9/34)
- OSPF 패킷 (2/22)
 - 공통 헤더

필드	크기	역할
버전(Version)	8 비트	프로토콜 버전 정의
유형(Type)	8 비트	패킷 유형 표시(1~5)
메시지 길이(Message length)	16 비트	헤더를 포함한 전체 메시지 길이 표시
발신지 라우터 주소(Source router IP address)	32 비트	패킷을 송신한 라우터 IP주소 표시
지역 식별자(Area Identification)	32 비트	라우팅이 일어나는 지역 표시
검사합(Checksum)	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 않는 것을, 1은 패스워드 사용을 의미)
인증(Authentication)	32 비트	인증 데이터의 실제 값 (인증 종류가 0일 경우 0으로 채워지며, 1일 경우 8개 문자의 패스워드가 포함됨)



링크 상태 라우팅

- OSPF (Open Shortest Path First) (10/34)
- OSPF 패킷 유형 (3/22)
 - Hello 패킷 (1/3)
 - 정의
 - 이웃 관계를 생성하고 이웃의 도달 가능성을 검사하는 패킷
 - 특징
 - 링크 상태 라우팅의 첫 번째 과정
 - 주변 OSPF 라우터를 발견하고 이웃 관계를 형성
 - 일정 주기로 전송하여 인접 라우터의 동작 상태를 확인
 - 인접 라우터가 일정 시간 동안 Hello 패킷을 보내지 않으면 해당 라우터를 장애 상태로 판단



링크 상태 라우팅

- OSPF (Open Shortest Path First) (11/34)

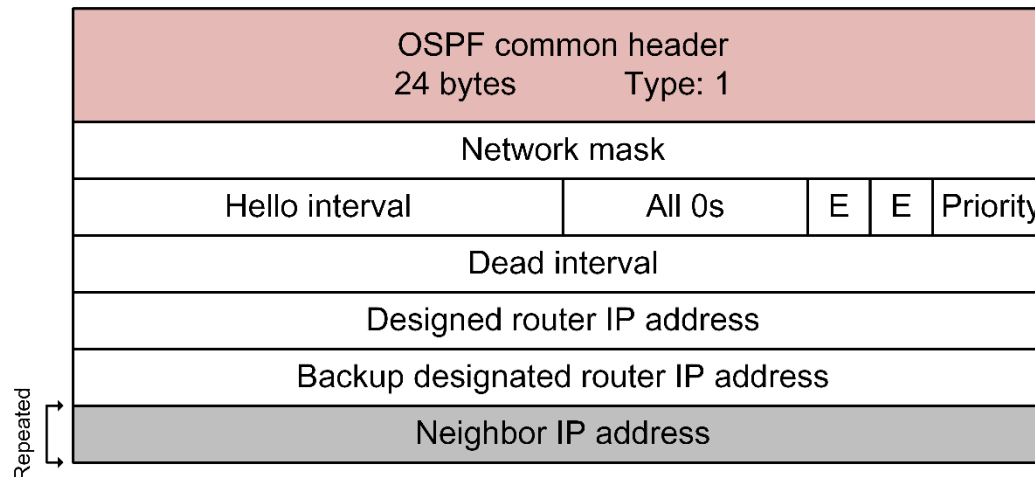
- OSPF 패킷 유형 (4/22)

- Hello 패킷 (2/3)

*스터브 지역(Stub Area)

외부 네트워크의 상세한 라우팅 정보를 차단하고
기본 경로를 사용하도록 구성한 OSPF 영역

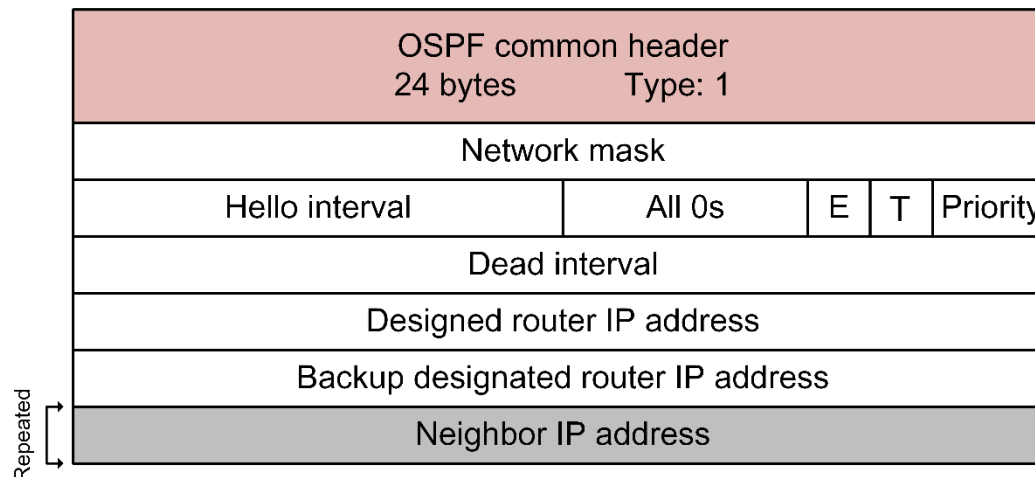
필드	크기	역할
네트워크 마스크(Network mask)	32 비트	hello 메시지가 전송되어지는 네트워크의 네트워크 마스크를 표시
Hello 간격(Hello interval)	16 비트	hello 메시지 간의 초 단위 설정
E 플래그	1 비트	1로 설정되어있으면 지역이 �터브 지역이 됨
T 플래그	1 비트	1로 설정되면 라우터는 다중 메트릭을 지원
우선순위(Priority)	6 비트	라우터의 우선순위를 표시 우선순위는 지정 라우터와 백업 지정 라우터를 결정



링크 상태 라우팅

- OSPF (Open Shortest Path First) (12/34)
 - OSPF 패킷 유형 (5/22)
 - Hello 패킷 (3/3)

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



링크 상태 라우팅

- OSPF (Open Shortest Path First) (13/34)

- OSPF 패킷 유형 (6/22)

*링크 상태 광고(LSA, Link State Advertisement)
라우터가 자신이 알고 있는 링크 상태를 이웃 라우터에게 알릴 때 사용하는 메시지

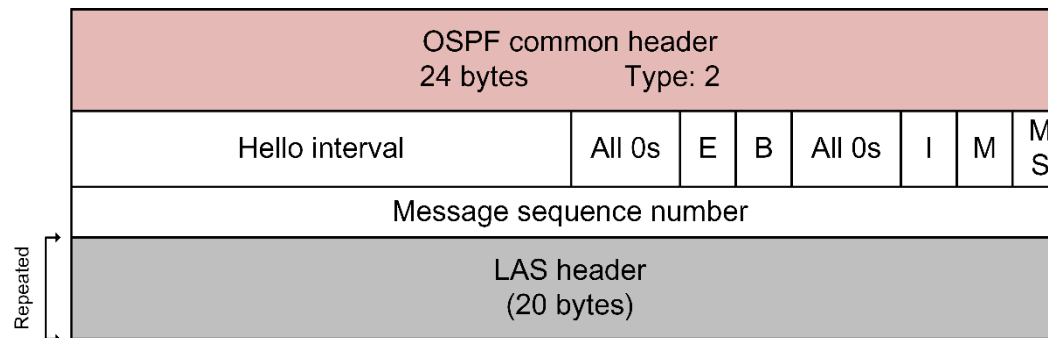
- 데이터베이스 기술(DBD, DataBase Description) 패킷 (1/2)

- 정의

- 두 OSPF 라우터가 서로 보유한 LSDB(Link-State DataBase)의 내용을 요약하여 교환하는 패킷

- 특징

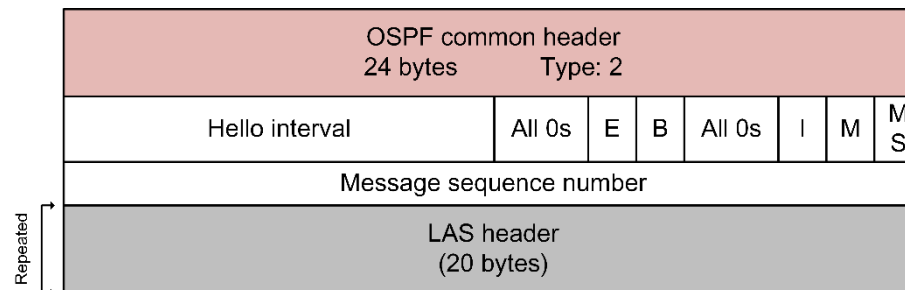
- 전체 링크 상태 정보를 전달하는 것이 아닌, LSA(Link State Advertisement)의 요약 정보를 전달
 - 상대 라우터가 어떤 LSA를 가지고 있는지 확인하는 데 사용
 - LSDB 동기화 과정에서 교환
 - 필요한 LSA를 확인한 후 이후 패킷을 통해 해당 정보를 요청



링크 상태 라우팅

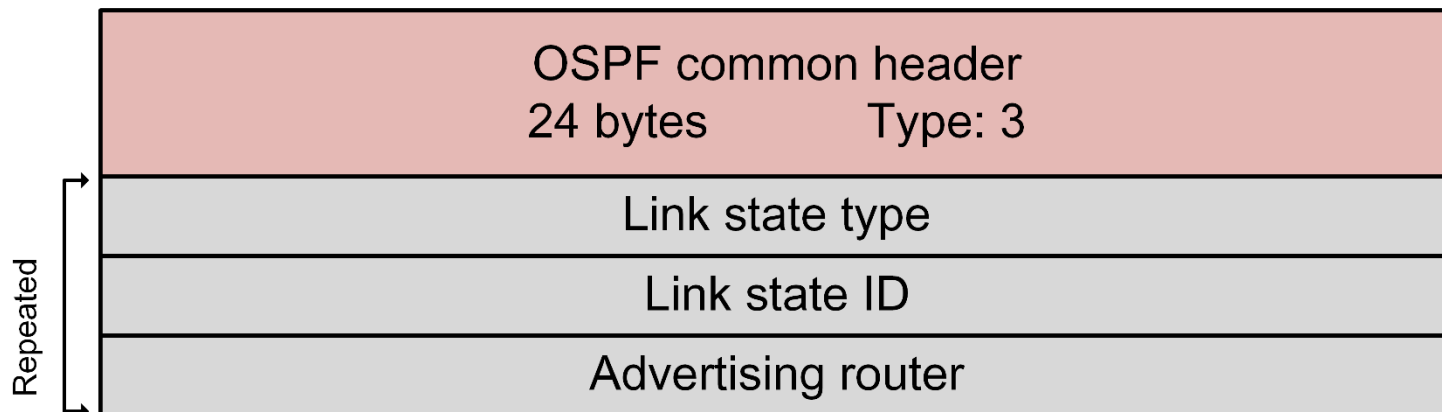
- OSPF (Open Shortest Path First) (14/34)
- OSPF 패킷 유형 (7/22)
 - 데이터베이스 기술(DBD, DataBase Description) 패킷 (2/2)

필드	크기	역할
E 플래그	1 비트	광고 라우터가 자율 시스템 경계 라우터이면 1로 지정
B 플래그	1 비트	광고 라우터가 지역 경계 라우터일 경우 1로 지정
I 플래그	1 비트	메시지가 첫 메시지일 때 1로 지정
M 플래그	1 비트	마지막 메시지가 아닐 경우에 1로 지정
M/S 플래그	1 비트	패킷의 시작점을 나타냄 1로 지정되면 주를, 0으로 지정되면 종임을 표시
메시지 순서 번호 (Message sequence number)	32 비트	메시지의 순서 번호를 포함하여 요청에 대한 응답 비교
LSA 헤더 (LSA header)	20 바이트	각 LSA에 사용 각 링크의 윤곽을 알려주며, 링크 상태 데이터베이스의 각 링크에 대해 반복



링크 상태 라우팅

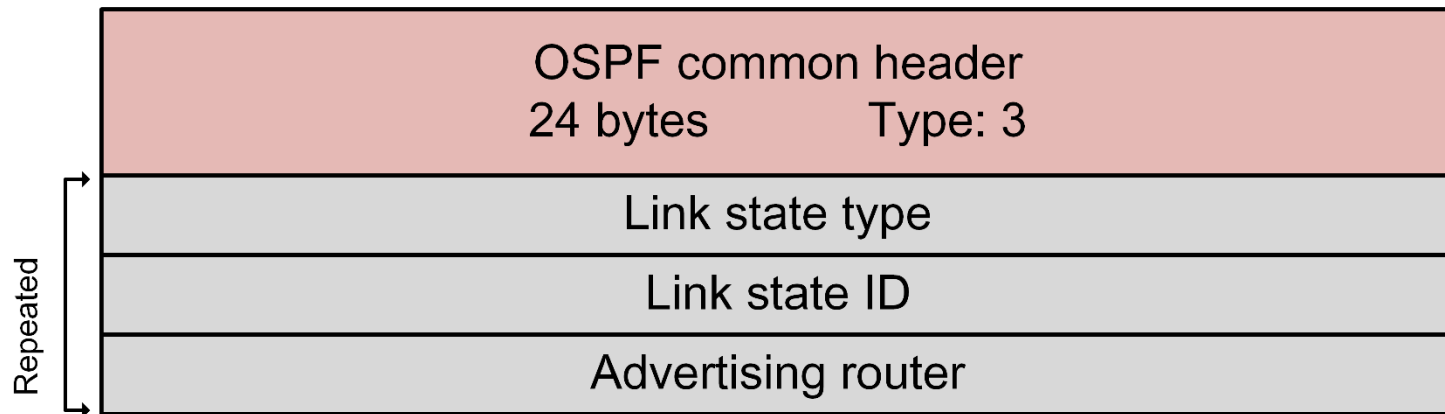
- OSPF (Open Shortest Path First) (15/34)
- OSPF 패킷 유형 (8/22)
 - 링크 상태 요청(LSR, Link State Request) 패킷 (1/2)
 - 정의
 - 상대 라우터가 가지고 있는 LSA 중 자신에게 필요한 정보를 요청하는 패킷
 - 특징
 - 데이터베이스 기술 패킷을 비교한 후 자신에게 없는 LSA를 확인하여 요청
 - 필요한 LSA만 선택적으로 요청 가능
 - LSDB를 동일한 상태로 동기화하기 위한 과정에서 사용



링크 상태 라우팅

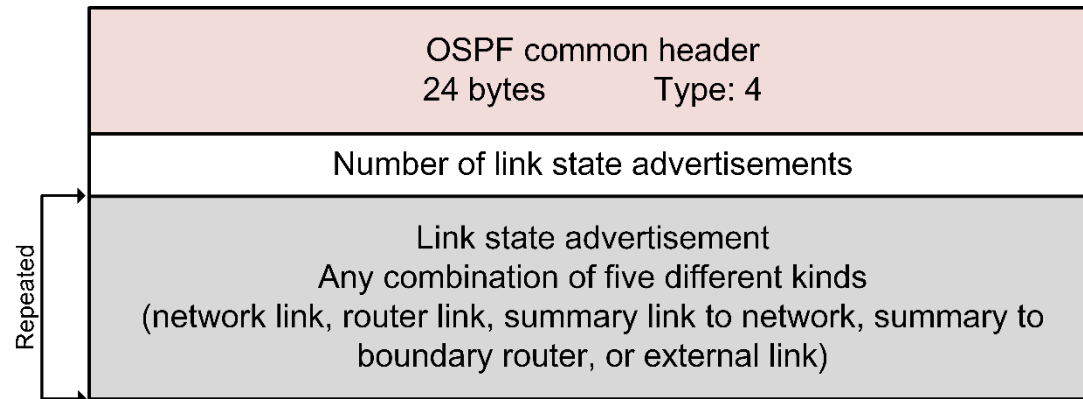
- OSPF (Open Shortest Path First) (16/34)
- OSPF 패킷 유형 (9/22)
 - 링크 상태 요청(LSR, Link State Request) 패킷 (2/2)

필드	크기	역할
링크 상태 유형(Link state type)	4 바이트	원하는 LSA 유형
링크 상태 ID(Link state ID)	4 바이트	LSA 식별을 위해 연결된 라우터 또는 네트워크의 IP 주소
광고 라우터(Advertising router)	4 바이트	갱신 요청된 LSA를 생성해야 하는 라우터 ID



링크 상태 라우팅

- OSPF (Open Shortest Path First) (17/34)
- OSPF 패킷 유형 (10/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (1/12)
 - 정의
 - 하나 이상의 LSA를 실제로 전달하는 패킷
 - 특징
 - 각 갱신 패킷은 몇 개의 서로 다른 LSA 포함 가능
 - LSR 패킷을 통해 요청 받은 LSA를 전달
 - 네트워크의 링크 상태가 변경되었을 때 새로운 LSA 전달에 사용
 - 수신한 LSA를 LSDB에 반영하여 네트워크 상태를 최신화



링크 상태 라우팅

- OSPF (Open Shortest Path First) (18/34)
 - OSPF 패킷 유형 (11/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (2/12)
 - LSA 일반적인 헤더

필드	크기	역할
링크 상태 시간 (Link state age)	16 비트	메시지 생성 후 경과된 시간(초 단위) LSA 수명과 최신 여부를 판단하는데 사용
E 플래그	1 비트	스터브 지역일 경우 1로 지정
T 플래그	1 비트	라우터가 여러 개의 서비스 유형을 처리할 수 있으면 1로 지정
링크 상태 유형 (Link state type)	7	LSA 유형을 의미 (1) - 라우터 링크, (2) - 네트워크 링크, (3) - 네트워크 요약 링크, (4) - 자율 시스템 경계 라우터의 요약 링크, (5) - 외부 링크
링크 상태 식별자 (Link state ID)	32 비트	링크의 유형에 의존하여 결정 (1) - 라우터 링크, (2) - 네트워크 링크, (3) - 네트워크 요약 링크, (4) - 자율 시스템 경계 라우터의 요약 링크, (5) - 외부 링크

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

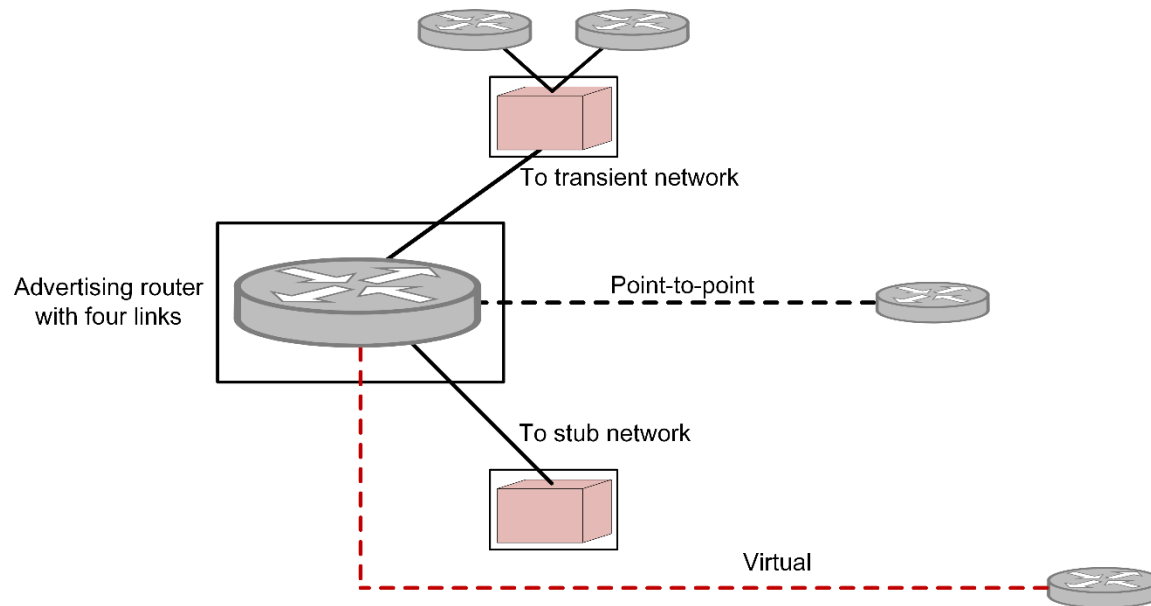
- OSPF (Open Shortest Path First) (19/34)
- OSPF 패킷 유형 (12/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (3/12)
 - LSA 일반적인 헤더

필드	크기	역할
광고 라우터 (Advertising router)	32 비트	이 메시지를 광고하는 라우터의 IP 주소
링크 상태 순서 번호 (Link state sequence number)	32 비트	각 링크 상태 갱신 메시지에 할당된 순서 번호
링크 상태 검사합 (Link state checksum)	16 비트	경과 시간 필드를 제외한 전체 패킷의 폴레처의 검사합이라는 특별한 검사합 계산 방법 사용
길이(Length)	16 비트	전체 패킷의 길이를 바이트 단위로 나타냄

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) (20/34)
- OSPF 패킷 유형 (13/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (4/12)
 - 라우터 링크 LSA
 - 라우터(실제 라우터)의 모든 링크들에 관해 광고
 - 라우터는 이 LSA를 사용하여 라우터의 모든 링크들에 관해 광고

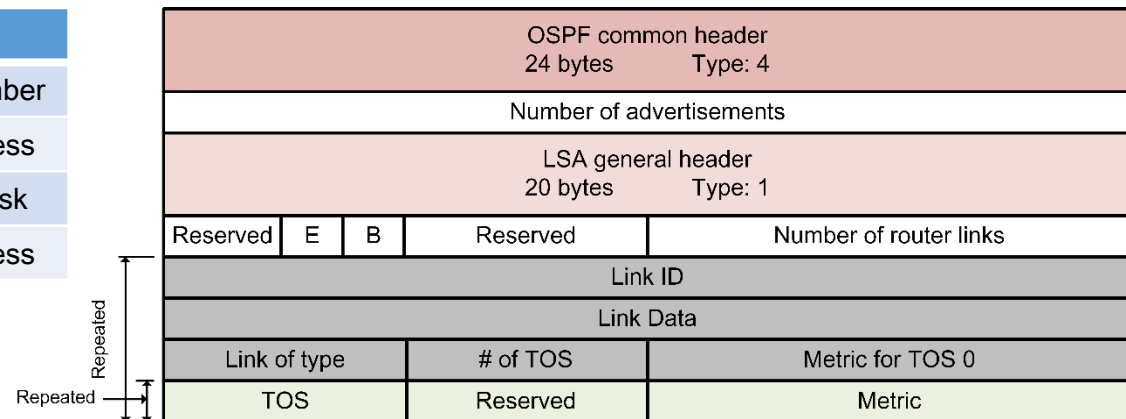


링크 상태 라우팅

- OSPF (Open Shortest Path First) (21/34)
- OSPF 패킷 유형 (14/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (5/12)
 - 라우터 링크 LSA

필드	크기	역할
링크 식별자(Link ID)	32 비트	이 메시지를 광고하는 라우터의 IP 주소
링크 데이터(Link data)	32 비트	링크에 관한 추가 정보 제공 링크 유형에 의존한 값을 지님
링크 유형(Link type)	8 비트	라우터가 연결된 네트워크의 유형에 따라 4가지의 서로 다른 유형의 링크 유형이 정의됨

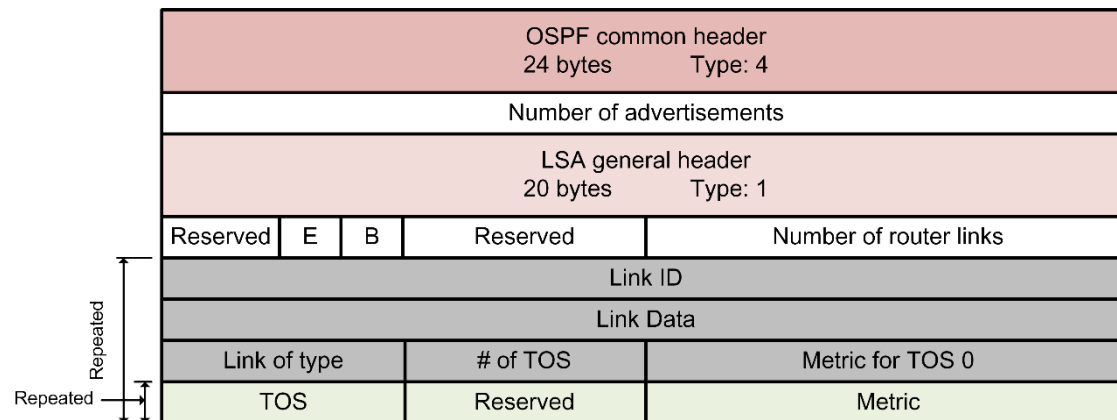
Link Type	Link Identification	Link Data
Type1: Point-to-point	Address of neighbor router	Interface number
Type2: Transient	Address of designated router	Router address
Type3: Stub	Network address	Network mask
Type4: Virtual	Address of neighbor router	Router address



링크 상태 라우팅

- OSPF (Open Shortest Path First) (22/34)
- OSPF 패킷 유형 (15/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (6/12)
 - 라우터 링크 LSA

필드	크기	역할
서비스 유형(TOS)의 개수	16 비트	각 링크에 알려진 서비스 유형의 개수를 나타냄
TOS 0에 대한 메트릭 (Metric for TOS 0)	8 비트	기본 서비스인 TOS 0에 대한 메트릭을 나타냄
TOS(Type Of Service)	8 비트	서비스 유형을 나타냄
메트릭(Metric)	16 비트	해당 TOS에 대한 메트릭을 나타냄

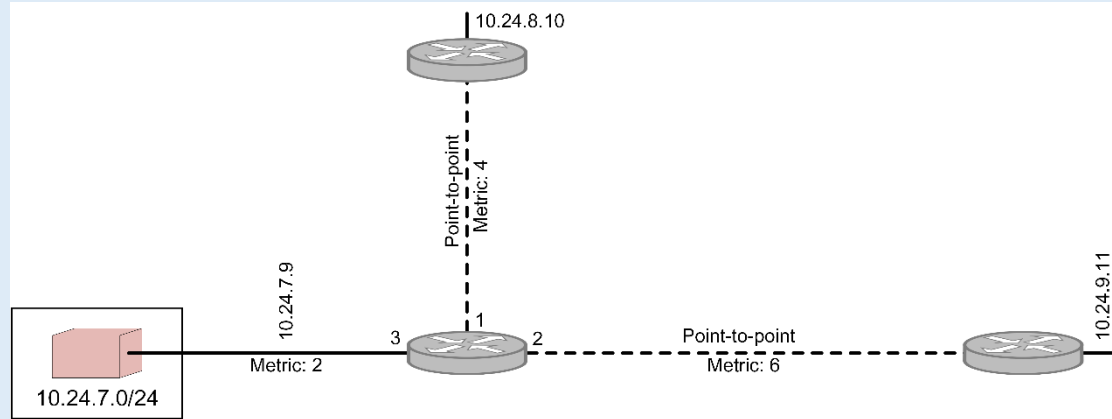


링크 상태 라우팅

• OSPF (Open Shortest Path First) (23/34)

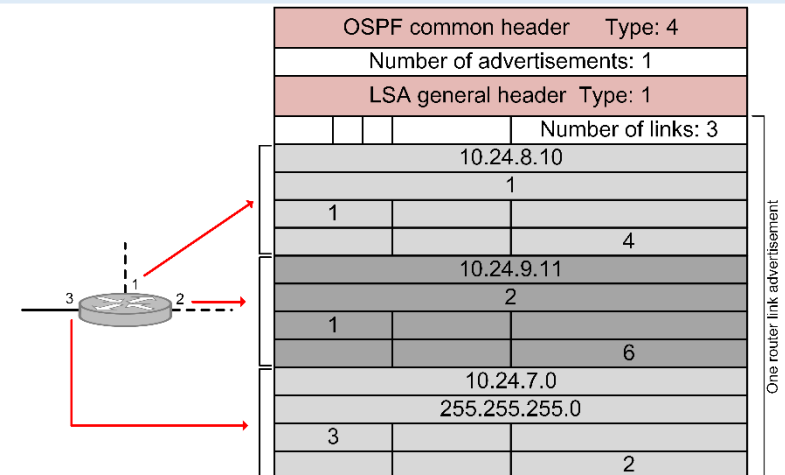
• 예제 11.4

아래 그림을 바탕으로, 라우터 10.24.7.9에 의해 전송되는 라우터 링크 LSA를 나타내어라.



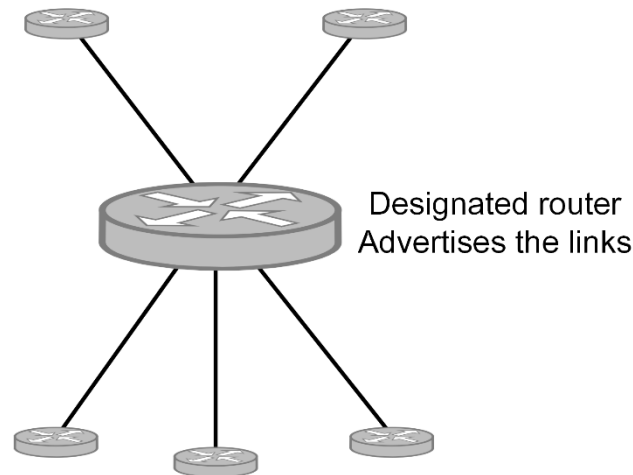
• 풀이

- 라우터는 3개의 링크를 가짐
- 이 중 둘은 Type 1(Point-to-point)이며, 하나는 Type 3(Stub Network)
- Stub Network에서는 Link Data 부분에 Network mask가 들어가므로 255.255.255.0
- 각각의 Metric은 4, 6, 2



링크 상태 라우팅

- OSPF (Open Shortest Path First) (24/34)
- OSPF 패킷 유형 (16/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (7/12)
 - 네트워크 링크 LSA
 - 네트워크의 링크들 정의
 - 지정 라우터는 경유 망을 대신 해 이런 유형의 LSP 패킷들을 분배
 - 네트워크에 연결된 모든 라우터들에게 존재를 알림

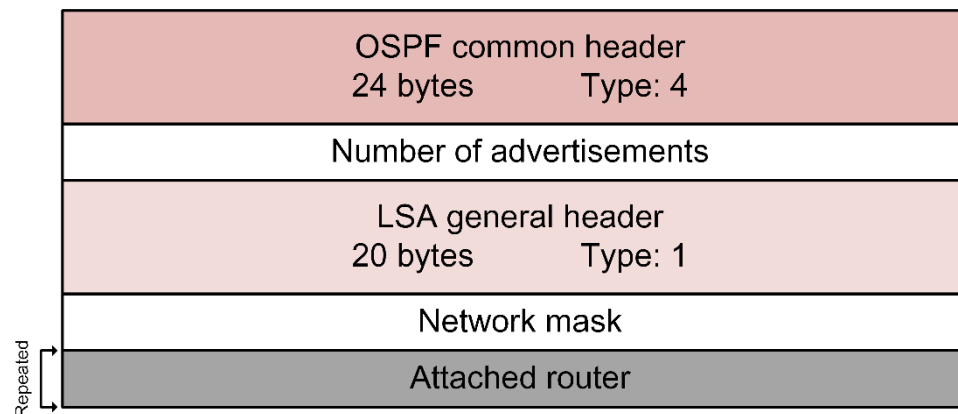


Network with five links

링크 상태 라우팅

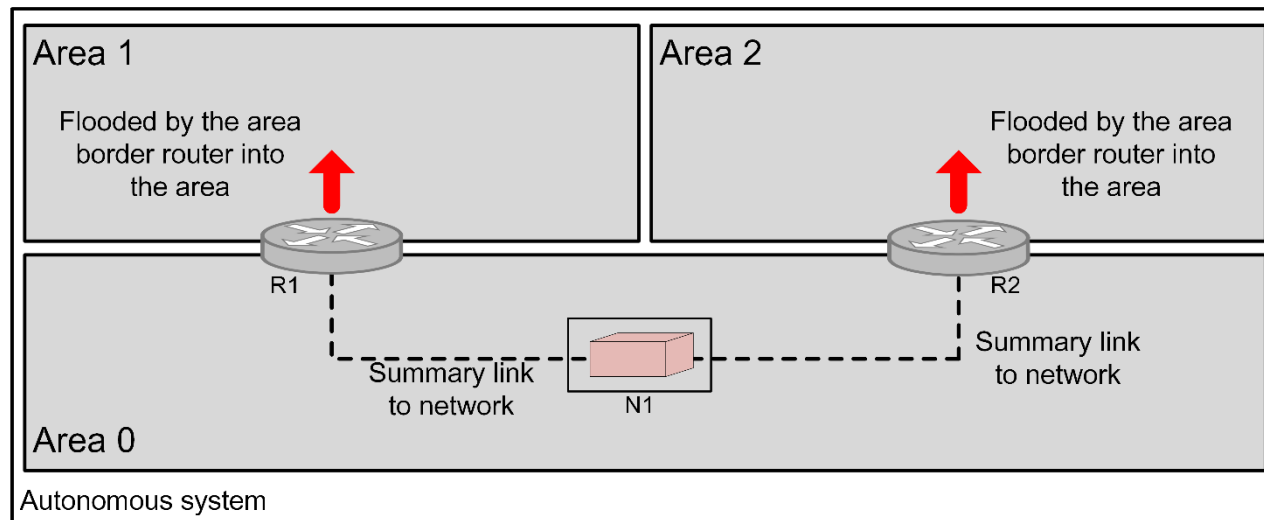
- OSPF (Open Shortest Path First) (25/34)
- OSPF 패킷 유형 (17/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (8/12)
 - 네트워크 링크 LSA

필드	크기	역할
네트워크 마스크 (Network mask)	32 비트	네트워크 마스크를 나타냄
접속 라우터 (Attached router)	32 비트	반복되는 필드로서 네트워크에 접속된 모든 라우터의 IP 주소들을 나타냄



링크 상태 라우팅

- OSPF (Open Shortest Path First) (26/34)
- OSPF 패킷 유형 (18/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (9/12)
 - 네트워크 요약 링크 LSA
 - 지역 바깥의 다른 망 존재를 알리기 위해 지역 경계 라우터에 의해 사용되는 LSA
 - 네트워크 마스크와 각 서비스 종류에 대한 메트릭으로 구성
 - 각 광고는 하나의 단일 네트워크 만을 알림
 - 광고하는 라우터의 IP 주소는 링크 상태 광고의 헤더에 포함되어 알려짐

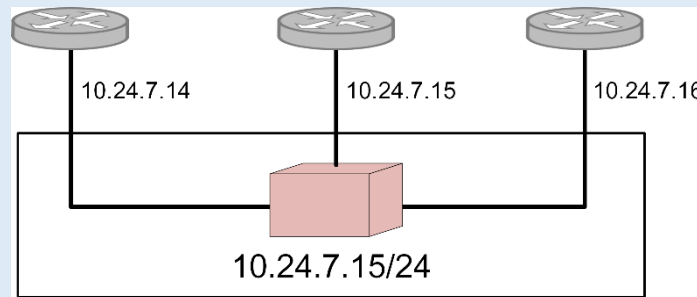


링크 상태 라우팅

- OSPF (Open Shortest Path First) (27/34)

- 예제 11.5

아래 그림을 바탕으로, 네트워크 링크 LSA를 나타내어라.



- 풀이

- 네트워크 링크를 광고하는 네트워크에는 3개의 라우터가 연결
- LSA는 마스크와 라우터의 주소를 보여줌

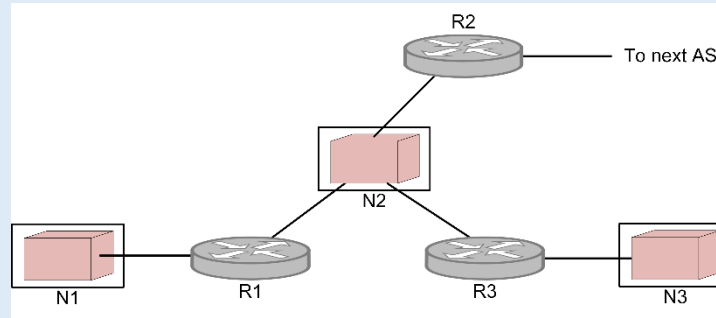
OSPF common header	Type: 4
Number of advertisements: 1	
LSA general header	Type: 2
255.255.255.0	
10.24.7.14	
10.24.7.15	
10.24.7.16	

링크 상태 라우팅

- OSPF (Open Shortest Path First) (28/34)

- 예제 11.6

아래 그림 바탕에서, 어느 라우터가 라우터 링크 LSA를 전송하는가?



- 풀이

- a. R1은 N1과 N2의 두 개의 링크를 가짐
- b. R2는 N1 하나의 링크만을 가짐
- c. R3은 N2 및 N3 두 개의 링크를 가짐

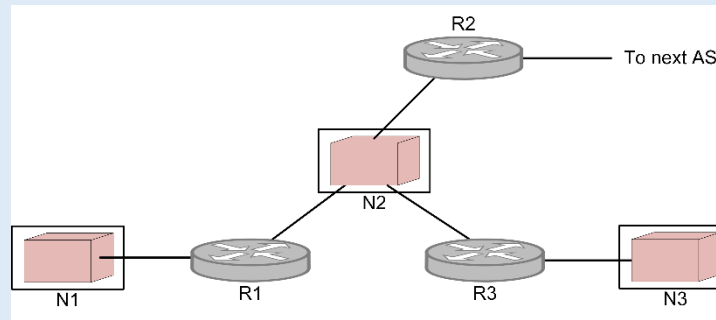
- 따라서, 모든 라우터가 라우터 링크 LSA들을 광고함

링크 상태 라우팅

- OSPF (Open Shortest Path First) (29/34)

- 예제 11.7

아래 그림에서, 어느 라우터가 네트워크 링크 LSA를 광고하는가?



- 풀이

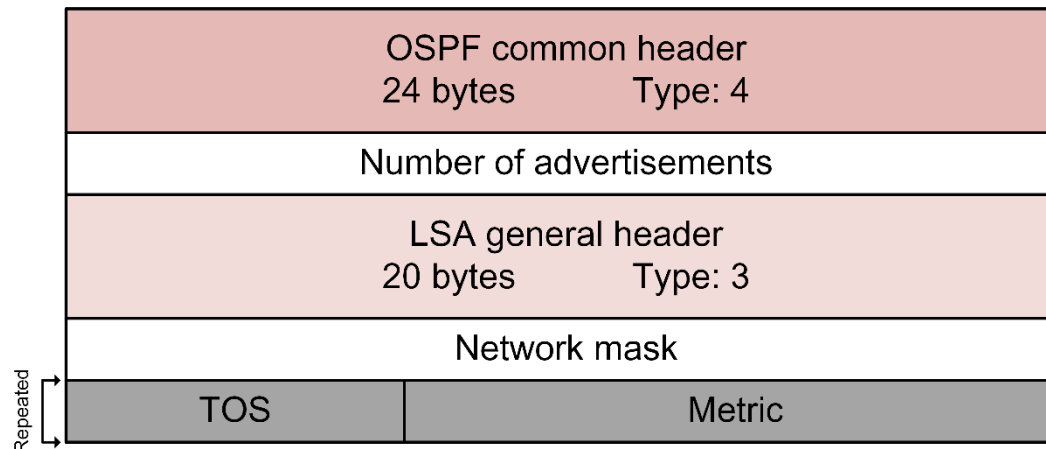
- N1에 대한 광고는 R1이 수행
 - 이는 자신만이 접속 라우터이고 지정 라우터가 됨
- N2에 대한 광고는 누가 지정 라우터로 선출되었는지에 따라 R1, R2 및 R3에 의해 가능할 수 있음
- N3에 대한 광고는 R3에 의해 수행
 - 이는 자신만이 접속 라우터이고 지정 라우터가 되기 때문

- 따라서, 모든 라우터가 네트워크 링크 LSA들을 광고함

링크 상태 라우팅

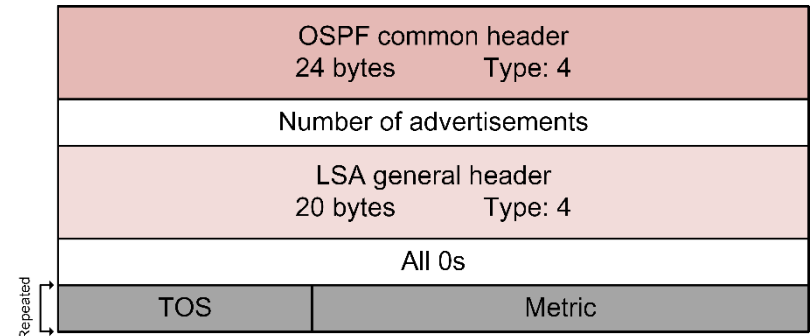
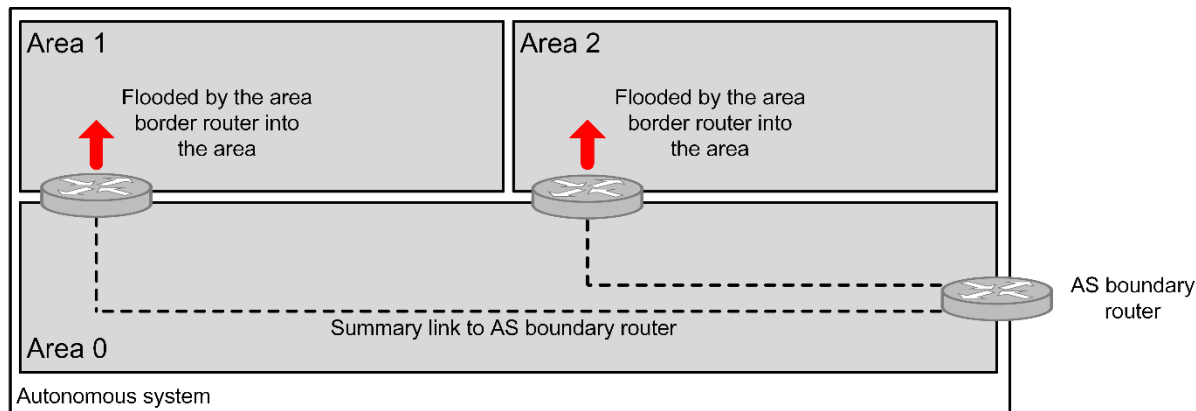
- OSPF (Open Shortest Path First) (30/34)
- OSPF 패킷 유형 (19/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (10/12)
 - 네트워크 요약 링크 LSA

필드	크기	역할
네트워크 마스크 (Network mask)	32 비트	네트워크 마스크를 나타냄
TOS	8 비트	서비스의 유형을 나타냄
메트릭(Metric)	24 비트	TOS 필드에서 정의된 서비스 유형에 대한 메트릭을 나타냄



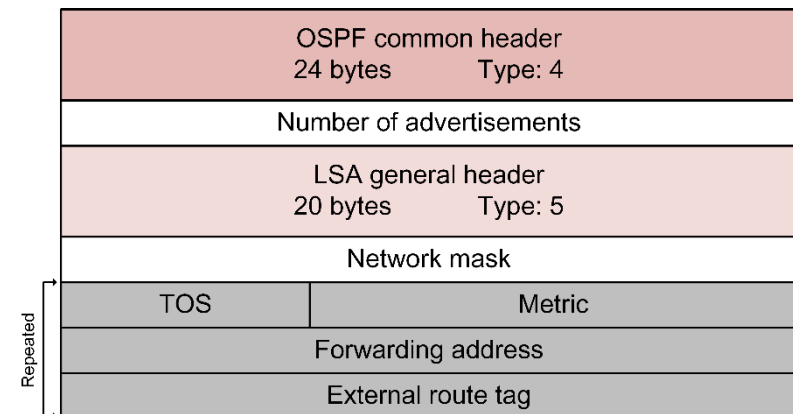
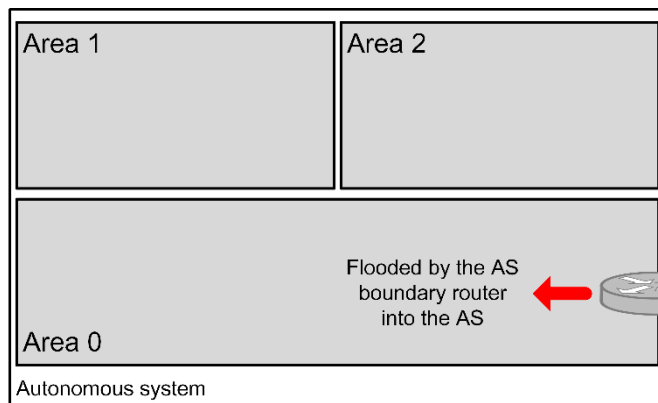
링크 상태 라우팅

- OSPF (Open Shortest Path First) (31/34)
 - OSPF 패킷 유형 (20/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (11/12)
 - 자율 시스템 경계 라우터로의 요약 링크 LSA
 - 자율 시스템 경계 라우터로의 경로를 제공하는 LSA
 - 지역 내의 라우터가 자율 시스템 바깥으로 패킷을 전송할 때 필요
 - 자율 시스템 외부로 전송될 메시지가 네트워크(e.g., Area 0)에 도달 시, 자율 경계 라우터는 해당 메시지를 수신
 - 패킷 형식은 네트워크로의 요약 링크 광고 메시지와 동일



링크 상태 라우팅

- OSPF (Open Shortest Path First) (32/34)
- OSPF 패킷 유형 (21/22)
 - 링크 상태 갱신(LSU, Link State Update) 패킷 (12/12)
 - 외부 링크 LSA(External Link LSA)
 - 자율 시스템 외부에 있는 모든 네트워크를 알리는 LSA
 - 자율 시스템 외부에 이용 가능한 네트워크가 공급할 때 사용
 - 각 광고는 하나의 네트워크만을 알림
 - 자율 시스템 경계 라우터로의 요약 링크 LSA에서 2개 필드 추가
 - 포워딩 주소(Forwarding address)
 - 외부 라우터 태그(External router tag)



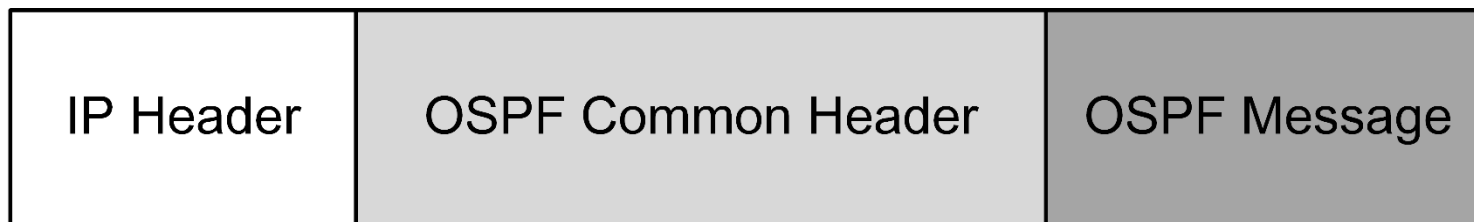
링크 상태 라우팅

- OSPF (Open Shortest Path First) (33/34)
- OSPF 패킷 유형 (22/22)
 - 링크 상태 확인응답(LSAck, Link State Acknowledgment) 패킷
 - 정의
 - 수신한 LSA가 정상적으로 전달되었음을 확인하는 패킷
 - 특징
 - LSU(Link State Update)로 전달받은 LSA의 수신을 확인
 - LSA의 신뢰성 있는 전달을 보장하는 데 사용
 - 확인 응답을 통해 LSA가 정상적으로 전달되었는지 판단 가능

OSPF common header	
24 bytes	Type: 5
LSA general header	
20 bytes	Corresponding type

링크 상태 라우팅

- OSPF (Open Shortest Path First) (34/34)
 - 캡슐화(Encapsulation)
 - OPSF는 IP 데이터그램에 캡슐화
 - OSPF 프로토콜은 흐름 제어와 오류 제어에 관한 응답 절차를 포함함
 - OSPF는 이 서비스를 제공하는 전송 계층 프로토콜을 필요로 하지 않음



목 차

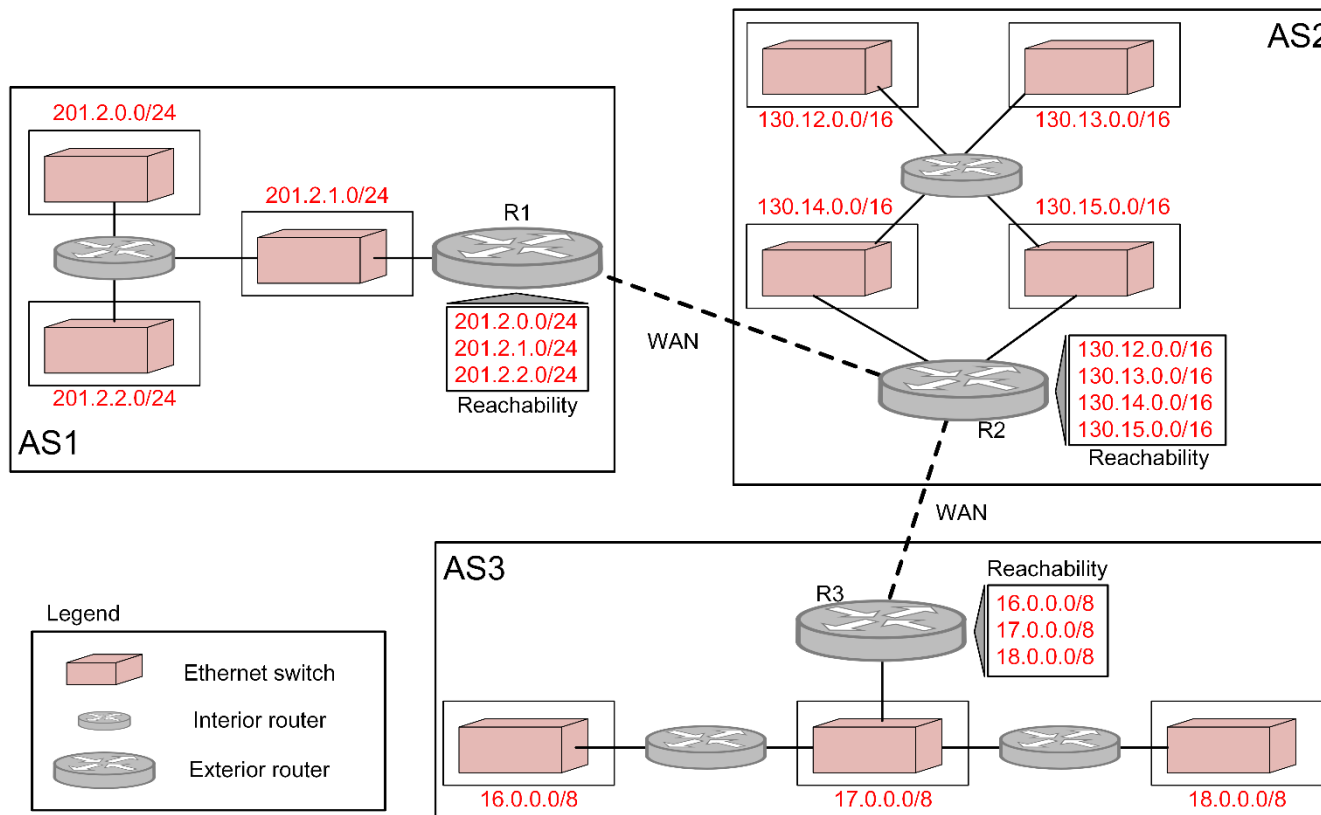
- 보충
- 도메인 내 및 도메인 간 라우팅
- 거리 벡터 라우팅
- 링크 상태 라우팅
- 경로 벡터 라우팅

경로 벡터 라우팅

- 경로 벡터 라우팅(Path Vector Routing) (1/5)
 - 정의
 - 목적지에 도달하기 위해 거쳐가야 하는 전체 경로의 목록을 공유하는 방식
 - 특징
 - 도메인 간 라우팅에 유용
 - 거리 벡터 라우팅과 유사
 - 스피커 라우터를 통해 인접 스피커 라우터와 정보 교환
 - 교환한 정보를 통해 테이블 재구성
 - BGP (Border Gateway Protocol)로 구현

경로 벡터 라우팅

- 경로 벡터 라우팅(Path Vector Routing) (2/5)
 - 도달 가능성(Reachability)
 - 각 AS에서 자신이 도달 가능한 네트워크 목록을 의미



경로 벡터 라우팅

- 경로 벡터 라우팅(Path Vector Routing) (3/5)
 - 라우팅 테이블 생성
 - 각 AS에 있는 스피커 라우터와 인접한 AS의 스피커 라우터에게 자신의 도달 가능성 목록을 공유하여 라우팅 테이블 갱신

R1



Network	Path
201.2.0.0/24	AS1 (This AS)
201.2.1.0/24	AS1 (This AS)
201.2.2.0/24	AS1 (This AS)
130.12.0.0/16	AS1, AS2
130.13.0.0/16	AS1, AS2
130.14.0.0/16	AS1, AS2
130.15.0.0/16	AS1, AS2
16.0.0.0/8	AS1, AS2, AS3
17.0.0.0/8	AS1, AS2, AS3
18.0.0.0/8	AS1, AS2, AS3

Path-vector Routing Table

R2



Network	Path
201.2.0.0/24	AS2, AS1
201.2.1.0/24	AS2, AS1
201.2.2.0/24	AS2, AS1
130.12.0.0/16	AS2, AS1
130.13.0.0/16	AS2 (This AS)
130.14.0.0/16	AS2 (This AS)
130.15.0.0/16	AS2 (This AS)
16.0.0.0/8	AS2, AS3
17.0.0.0/8	AS2, AS3
18.0.0.0/8	AS2, AS3

Path-vector Routing Table

R3



Network	Path
201.2.0.0/24	AS3, AS2, AS1
201.2.1.0/24	AS3, AS2, AS1
201.2.2.0/24	AS3, AS2, AS1
130.12.0.0/16	AS3, AS2
130.13.0.0/16	AS3, AS2
130.14.0.0/16	AS3, AS2
130.15.0.0/16	AS3, AS2
16.0.0.0/8	AS3 (This AS)
17.0.0.0/8	AS3 (This AS)
18.0.0.0/8	AS3 (This AS)

Path-vector Routing Table

경로 벡터 라우팅

- 경로 벡터 라우팅(Path Vector Routing) (4/5)
 - 라우팅 테이블 특징 (1/2)
 - 루프 방지
 - 거리 벡터 라우팅에서 일어나는 불안정성과 루프의 생성은 경로 벡터 라우팅에서 해결될 수 있음
 - 라우터가 메시지를 수신하면 목적지까지 경로 목록에 해당 자율 시스템이 있는지 검토
 - 만약 있다면, 루프가 일어난 것처럼 메시지를 무시
 - 정책 라우팅
 - 라우터가 메시지를 수신하면 경로를 검사하여 경로 상에 자율 시스템들 중 하나가 정책을 위반하는지 판단
 - 위반할 경우, 해당 경로에 대한 라우팅 테이블을 갱신하지도 않고 이웃에 전송하지 않음

경로 벡터 라우팅

• 경로 벡터 라우팅(Path Vector Routing) (5/5)

• 라우팅 테이블 특징 (2/2)

• 집적화

- 여러 네트워크의 경로 정보를 하나로 묶어 라우팅 테이블의 크기를 감소시키는 것

• 특징

- 일반적으로 CIDR 표기 가능 시, 주소의 집적화 가능
- 경로 정보를 단순화하여 라우팅 처리를 효율화하고 정보 교환 속도를 향상

*CIDR(Classless Inter-Domain Routing)
클래스 없는 도메인 간 라우팅의 약자

R1		R2		R3	
Network	Path	Network	Path	Network	Path
201.2.0.0/24	AS1 (This AS)	201.2.0.0/24	AS2, AS1	201.2.0.0/24	AS3, AS2, AS1
201.2.1.0/24	AS1 (This AS)	201.2.1.0/24	AS2, AS1	201.2.1.0/24	AS3, AS2, AS1
201.2.2.0/24	AS1 (This AS)	201.2.2.0/24	AS2, AS1	201.2.2.0/24	AS3, AS2, AS1
130.12.0.0/16	AS1, AS2	130.12.0.0/16	AS2, AS1	130.12.0.0/16	AS3, AS2
130.13.0.0/16	AS1, AS2	130.13.0.0/16	AS2, AS1	130.13.0.0/16	AS3, AS2
130.14.0.0/16	AS1, AS2	130.14.0.0/16	AS2, AS1	130.14.0.0/16	AS3, AS2
130.15.0.0/16	AS1, AS2	130.15.0.0/16	AS2, AS1	130.15.0.0/16	AS3, AS2
16.0.0.0/8	AS1, AS2, AS3	16.0.0.0/8	AS2, AS3	16.0.0.0/8	AS3, AS2
17.0.0.0/8	AS1, AS2, AS3	17.0.0.0/8	AS2, AS3	17.0.0.0/8	AS3, AS2
18.0.0.0/8	AS1, AS2, AS3	18.0.0.0/8	AS2, AS3	18.0.0.0/8	AS3, AS2

R1		R2		R3	
Network	Path	Network	Path	Network	Path
201.2.0.0/24	AS1 (This AS)	201.2.0.0/24	AS2, AS1	201.2.0.0/24	AS3, AS2, AS1
130.12.0.0/16	AS1, AS2	130.12.0.0/16	AS2, AS1	130.12.0.0/16	AS3, AS2
16.0.0.0/6	AS1, AS2, AS3	16.0.0.0/6	AS2, AS3	16.0.0.0/6	AS3, AS2

경로 벡터 라우팅

- BGP (Border Gateway Protocol) (1/18)

- 정의

- 서로 다른 자율 시스템(AS) 간에 라우팅 정보를 교환하고 경로를 결정하는 경계 게이트웨이 프로토콜

- 특징

- 경로 벡터 프로토콜
- 정책 기반 라우팅

- 단순 최단 경로가 아닌 관리자의 라우팅 정책에 따라 경로 선택
 - e.g., 특정 통신사 경로 우선 사용

- TCP 사용

- TCP 179번 포트를 사용하여 BGP 간 신뢰성 있는 통신

- 확장성

- 인터넷과 같이 매우 큰 규모의 네트워크에서도 라우팅 정보 교환 가능

경로 벡터 라우팅

- BGP (Border Gateway Protocol) (2/18)

- 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

- 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 (Border Gateway Protocol) (3/18)
 - 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 (Border Gateway Protocol) (4/18)

- 경로 속성

- 잘 알려진 속성

- BGP 라우터가 반드시 인식해야만 하는 속성

- 필수적인 속성

- 경로를 기술하는 데 반드시 있어야 하는 속성

- 임의의 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 (Border Gateway Protocol) (5/18)

• 세션

• 정의

- 라우팅 정보 교환을 위해서만 두 BGP 라우터들 간에 설정되는 연결

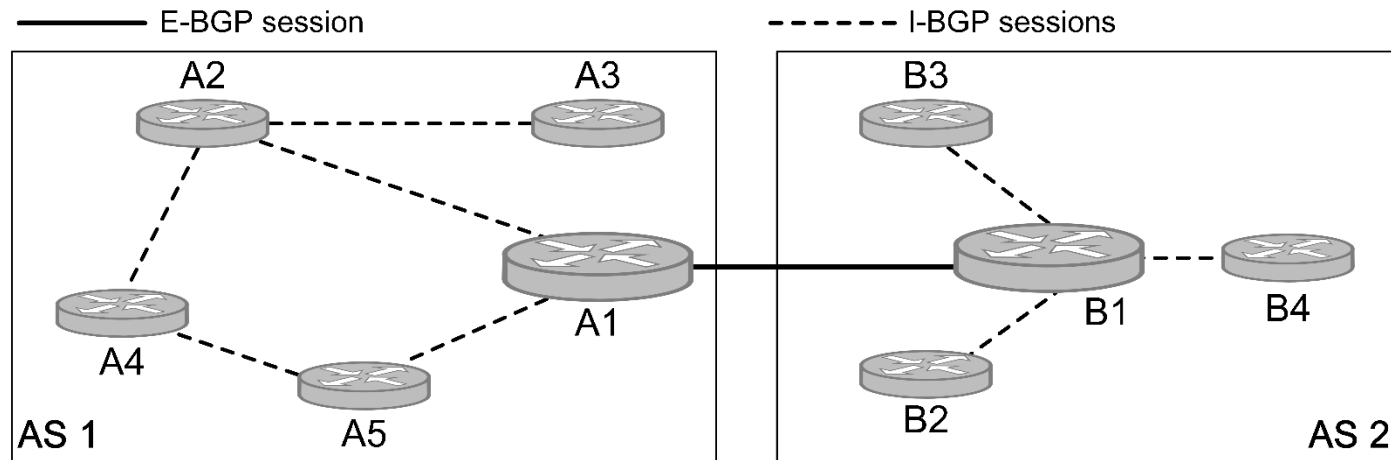
• 종류

- 외부(External) BGP 세션(E-BGP)

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

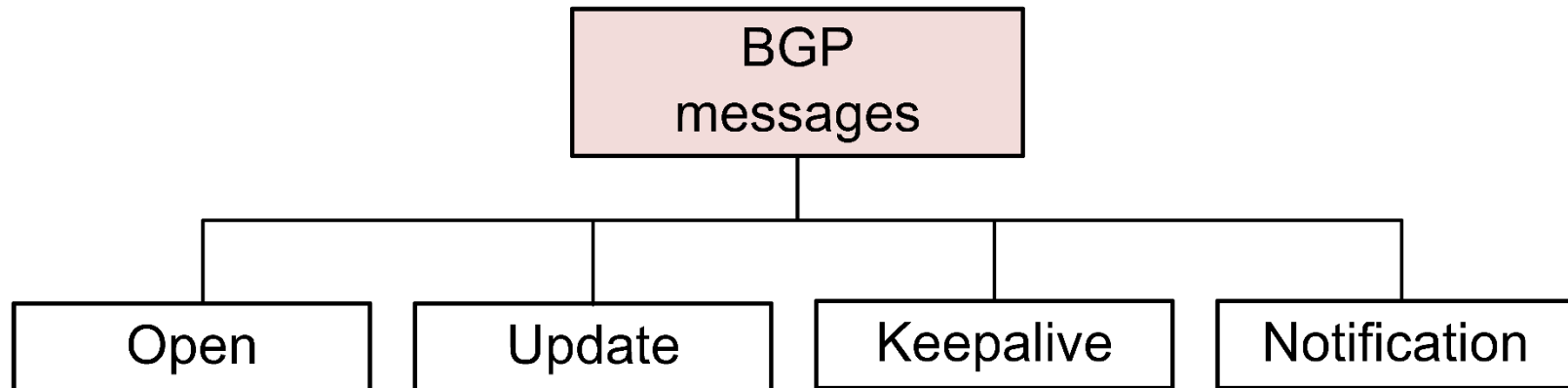
- 내부(Internal) BGP 세션(I-BGP)

- 자율 시스템 내의 두 라우터 간에 정보 교환을 위해 사용



경로 벡터 라우팅

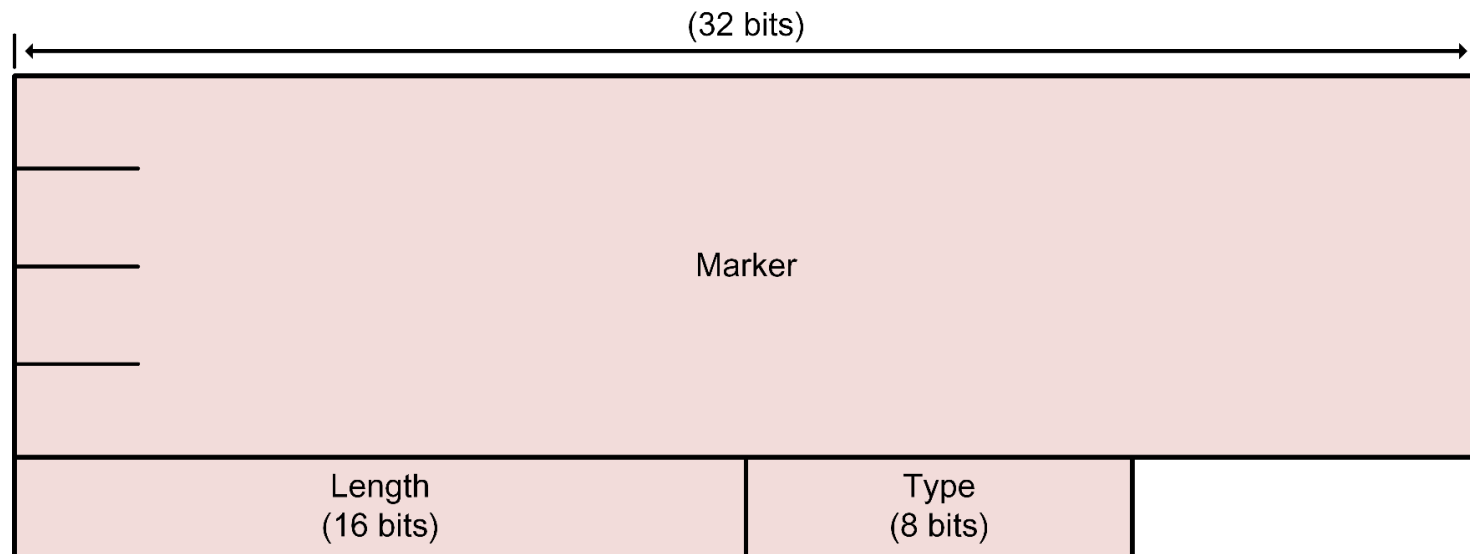
- BGP (Border Gateway Protocol) (6/18)
 - 패킷 종류
 - 개방(Open)
 - 갱신(Update)
 - 킵얼라이브(Keepalive)
 - 통지(Notification)



경로 벡터 라우팅

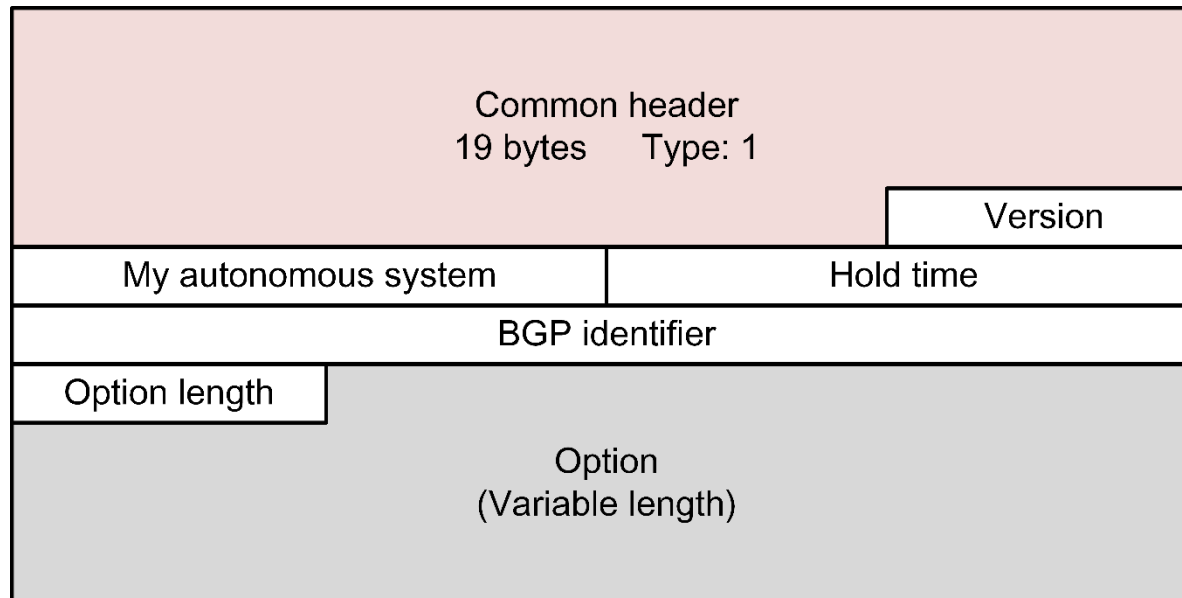
- BGP (Border Gateway Protocol) (7/18)
 - 패킷 형식 (1/11)
 - 헤더

필드	역할	크기
표시기(Marker)	인증을 위해 예약됨	256 비트
길이(Length)	헤더를 포함한 전체 메시지 길이	16 비트
유형(Type)	패킷의 유형 (개방, 갱신, 킵얼라이브, 통지)	8 비트



경로 벡터 라우팅

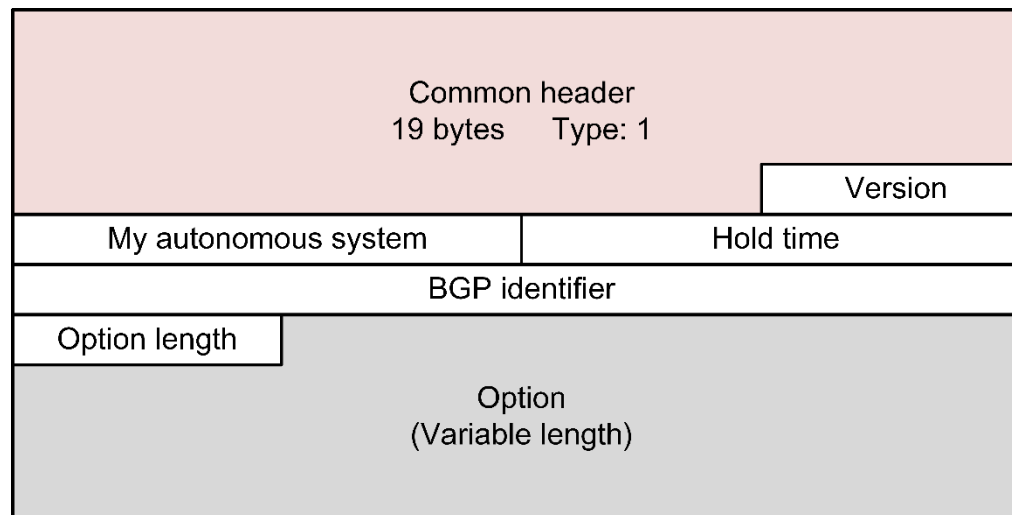
- BGP (Border Gateway Protocol) (8/18)
 - 패킷 형식 (2/11)
 - 개방(Open) (1/3)
 - BGP가 동작중인 라우터가 이웃 관계를 생성하기 위해서는 그 이웃과 TCP 연결을 열고 개방(Open) 메시지를 전송
 - 이웃이 관계를 받아들일 경우, 두 라우터 간에 이웃 관계가 성립되었다는 뜻으로 킵얼라이브(Keepalive) 메시지를 전송



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (9/18)
 - 패킷 형식 (3/11)
 - 개방(Open) (2/3)

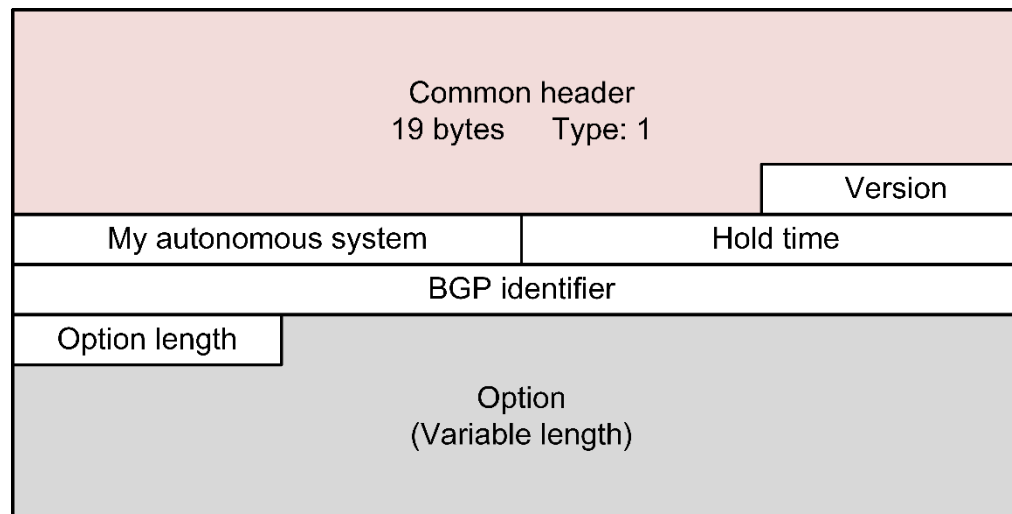
필드	역할	크기
버전(Version)	BGP의 버전을 표시(현재 4)	8 비트
자신의 자율 시스템 (My autonomous system)	자율 시스템 번호를 표시	16 비트
유지 시간(Hold time)	상대방으로부터 킵얼라이브나 갱신 메시지를 수신하기전까지 경과될 수 있는 시간(초 단위)	16 비트



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (10/18)
 - 패킷 형식 (4/11)
 - 개방(Open) (3/3)

필드	역할	크기
BGP 식별자 (BGP identifier)	개방 세미지를 전송한 라우터 IP 주소 표시	32 비트
선택 매개 변수 길이 (Option length)	전체 선택 매개 변수들의 길이 표시 (e.g., 인증, 멀티 프로토콜 확장 등)	8 비트
선택 매개 변수(Option)	각 선택 매개 변수들을 매개 변수 길이, 매개 변수 값의 서브 필드를 보유 (현재까지 표준화된 선택 매개 변수는 인증 뿐)	변동



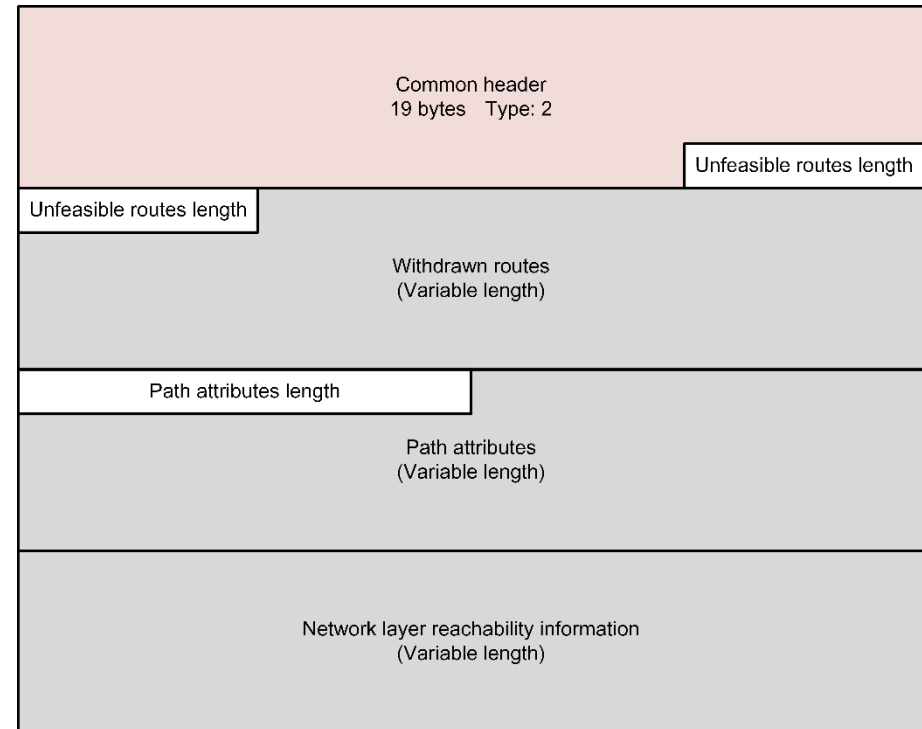
경로 벡터 라우팅

- BGP (Border Gateway Protocol) (11/18)

- 패킷 형식 (5/11)

- 갱신(Update) (1/4)

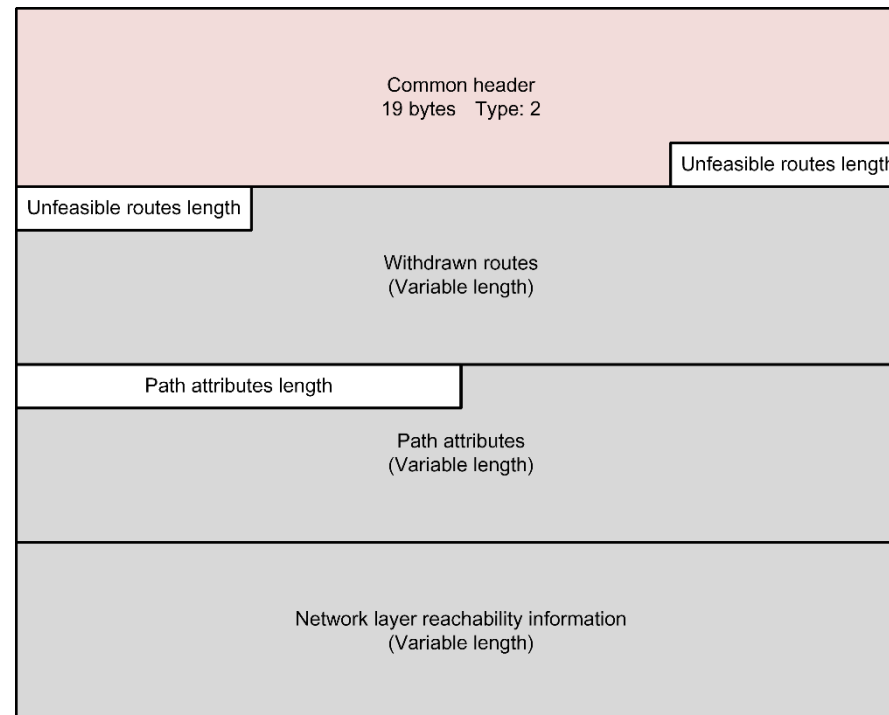
- BGP 프로토콜의 중심
 - 라우터로 하여금 전에 광고된 목적지를 취소하거나 새로운 목적지로의 경로를 알리는 데 사용
 - BGP는 전에 광고된 여러 목적지들을 취소할 수 있지만, 새로운 목적지를 광고할 경우 하나의 갱신 메시지를 통해 한 목적지만 가능



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (12/18)
 - 패킷 형식 (6/11)
 - 갱신(Update) (2/4)

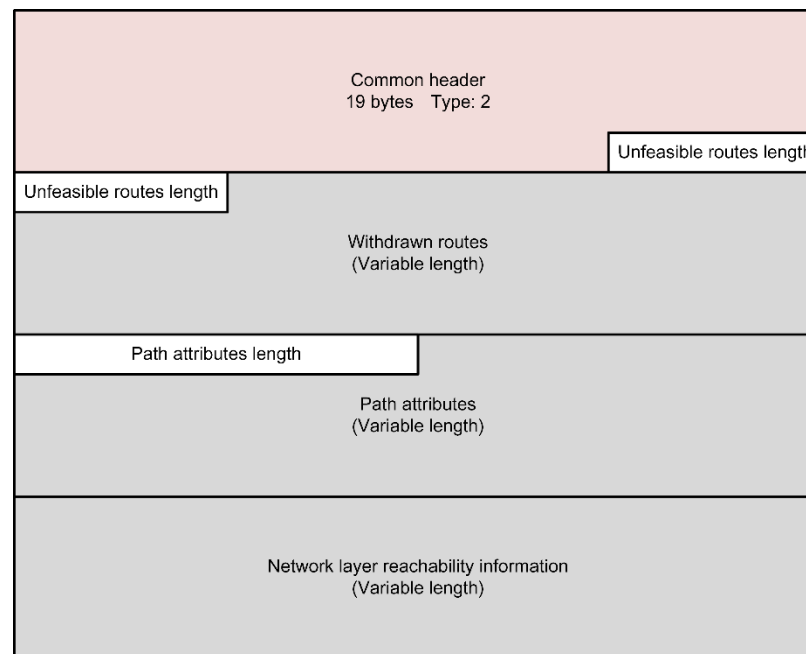
필드	역할	크기
불가능(Unfeasible) 경로 길이	다음 필드의 길이 정의	16 비트
취소된(Withdraw) 경로	전에 광고된 목록 중에서 삭제되어야 하는 모든 경로를 나열	변동



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (13/18)
 - 패킷 형식 (7/11)
 - 갱신(Update) (3/4)

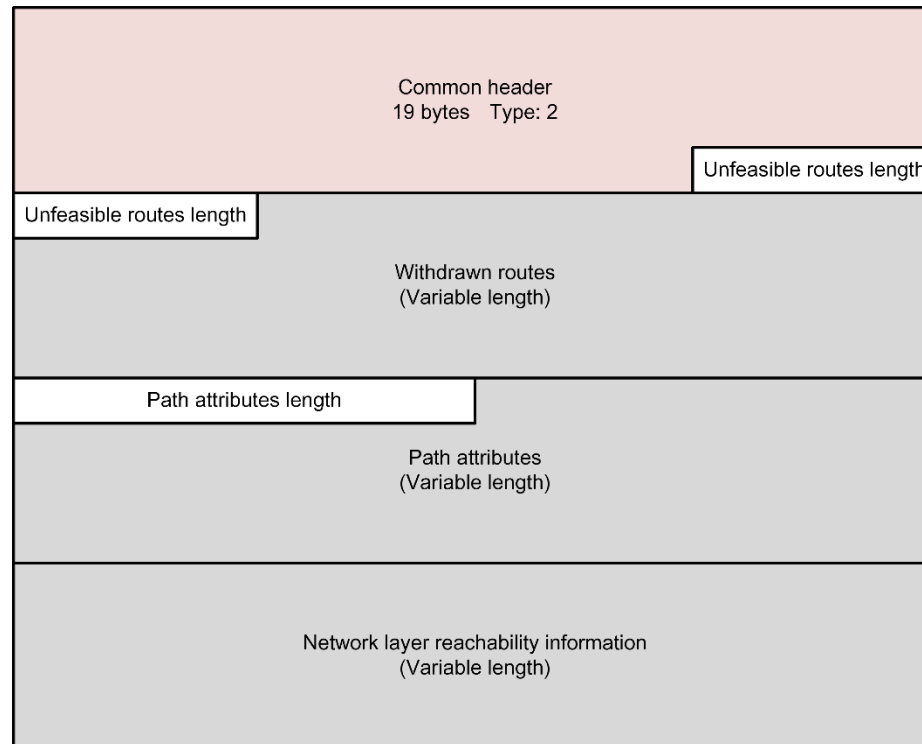
필드	역할	크기
경로 속성 길이 (Path attributes length)	다음 필드의 길이 정의	16 비트
경로 속성 (Path attributes)	이 메시지를 통해 광고되는 도달 가능한 네트워크까지 경로에 대한 속성을 나타냄	변동



경로 벡터 라우팅

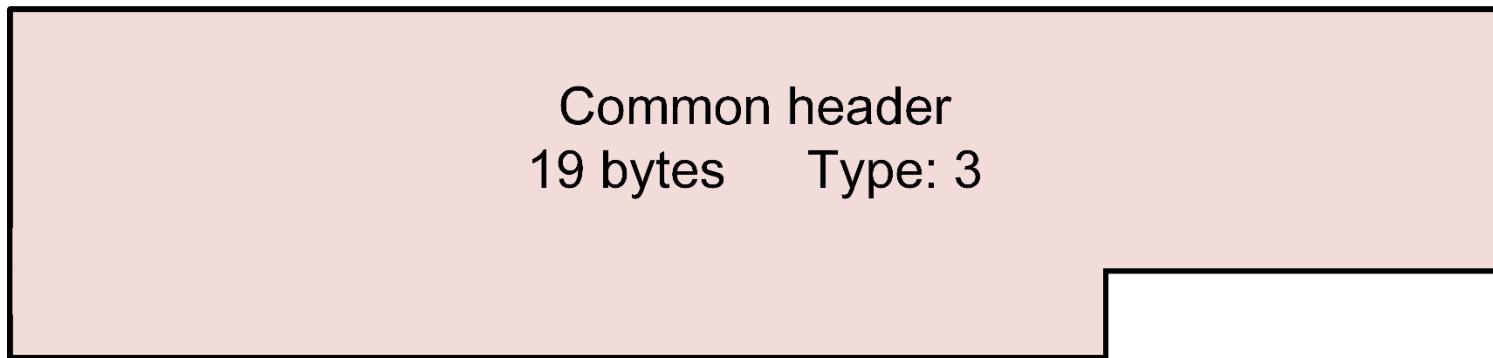
- BGP (Border Gateway Protocol) (14/18)
 - 패킷 형식 (8/11)
 - 갱신(Update) (4/4)

필드	역할	크기
네트워크 계층 도달 가능 정보 (Network layer reachability information)	메시지에 의해 실제로 광고되어지는 네트워크 표시 길이 필드와 IP주소 프리픽스를 보유	변동



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (15/18)
 - 패킷 형식 (9/11)
 - 킵얼라이브(Keepalive)
 - BGP가 동작하는 라우터들은 상대방에게 자신들이 살아있음을 알리기 위해 유지 시간이 만료되기 전에 정기적으로 킵얼라이브 메시지를 교환



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (16/18)

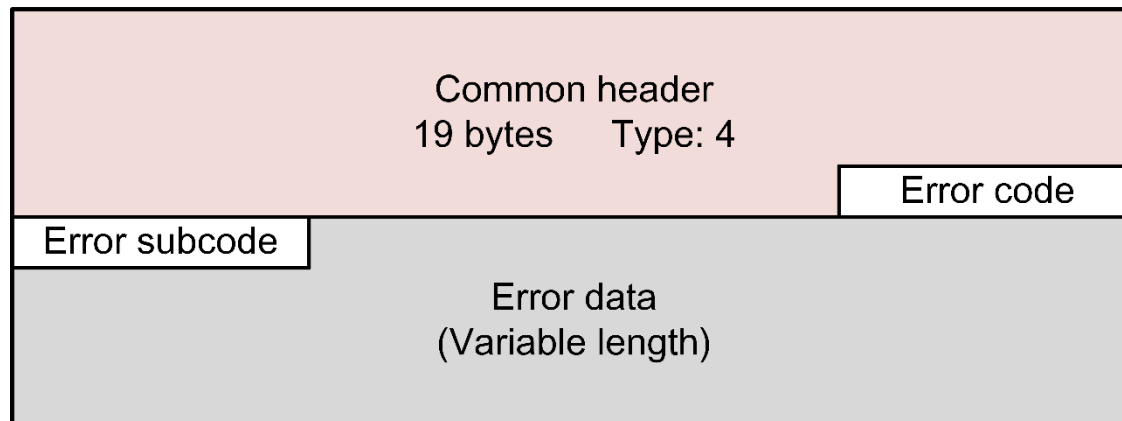
- 패킷 형식 (10/11)

- 통지(Notification) (1/2)

- 오류 상황이 감지되거나 연결을 닫기 원할 때 라우터에 의해 전송

- 메시지 형식

필드	역할	크기
오류 코드(Error code)	오류의 분류를 나타냄	8 비트
오류 서브 코드(Error subcode)	각 분류에서의 오류 유형을 나타냄	8 비트
오류 데이터(Error data)	오류에 대한 진단 정보를 제공하는데 사용	변동



경로 벡터 라우팅

- BGP (Border Gateway Protocol) (17/18)
 - 패킷 형식 (11/11)
 - 통지(Notification) (2/2)
 - 오류 코드

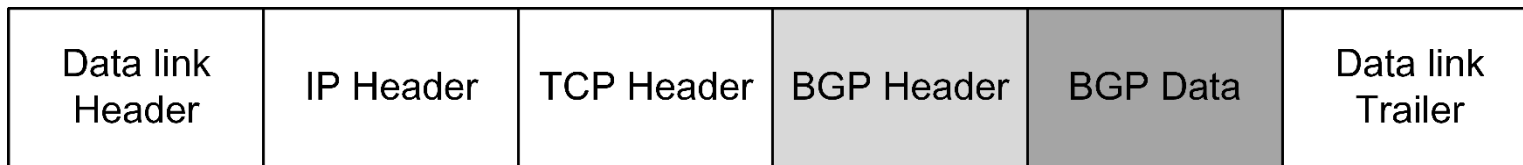
오류 코드	오류 코드 설명	오류 서브 코드 설명
1	메시지 헤더 오류	동기화 문제 (1), 잘못된 메시지 길이 (2), 잘못된 메시지 유형 (3)
2	개방 메시지 오류	지원되지 않는 버전 번호 (1), 잘못된 피어 AS (2), 잘못된 BGP 식별자 (3), 지원되지 않는 선택적 매개변수 (4), 인증 실패 (5), 허용되지 않는 Hold Time (6)
3	갱신 메시지 오류	잘못된 속성 목록 (1), 인식할 수 없는 Well-Known 속성 (2), 누락된 Well-Known 속성 (3), 속성 플래그 오류 (4), 속성 길이 오류 (5), 잘못된 Origin 속성 (6), AS 경로 루프 (7), 잘못된 Next Hop 속성 (8), Optional 속성 오류 (9), 잘못된 Network 필드 (10), 잘못된 AS_PATH (11)
4	유지 타이머 만료	-
5	유한 상태 기계 오류	절차 오류, 서브코드 없음
6	연결 종료	-

경로 벡터 라우팅

- BGP (Border Gateway Protocol) (18/18)

- 캡슐화

- 잘 알려진 포트 179를 사용하여 TCP 세그먼트에 캡슐화
 - 오류 제어나 흐름 제어가 불필요
 - 종료(Cease) 유형의 통지 메시지가 전송될 때까지 갱신(Update), 킵얼라이브(Keepalive), 통지(Notification) 메시지의 교환을 지속



Thanks!

서진우(jinwoo@pel.sejong.ac.kr)