

[CISCO Networking] part 7,8,9

[GBC20190027] Networking

7-1. 라우터



1. 라우터

- 지능을 가진 경로 배정기
- 라우터는 자신이 가야 할 길을 자동으로 찾아서 갈 수 있는 능력을 가진 것
- 외부의 어떤 인터넷 사이트를 찾아가는 데이터가 있다면 라우터는 이 데이터를 목적지까지 가장 빠르고 효율적인 길을 스스로 찾아 안내해 주는 능력

2. 세팅

- 네트워크에 대한 개념
- (네트워크 IP주소에 대한 이해, 서브넷 마스크에 대한 이해, 라우팅에 대한 이해)

3. 라우터가 하는 일

- IP 라우팅
- IPX, AppleTalk등 많은 프로토콜의 라우팅

7-2. 라우터는 무슨 일을 할까요?



1. 라우터가 하는 일

1) Path Determination (경로 결정)

- 라우터는 데이터 패킷이 목적지로 갈 수 있는 길을 검사하고 어떤 길로 가는 것이 가장 적절한지를 결정

2) Switching (스위칭)

- 경로 결정이 되면 그 쪽으로 데이터 패킷을 스위칭 해줌

2. 가장 좋은 경로 결정 방법

- 라우팅 알고리즘 사용 (라우팅 테이블을 만들어 관리)

7-3++. 라우터는 어떻게 생겼을까요?



- 라우터와 관련된 용어들

- 1) 인터페이스

- 라우터에 나와있는 접속 가능한 포트

ex. Cisco2501 : 라우터의 인터페이스는 1 이더넷과 2 시리얼이라고 함.
3개의 접속포트가 있다는 뜻

- 2) Ethernet

- 내부 네트워크와 접속 시에 사용하는 인터페이스
- 내부의 허브나 스위치 등과의 연결을 위한 포트

- 3) Serial

- WAN과의 접속을 위한 것
- DSU, CSU 등과의 연결
- 외부 네트워크로의 연결을 위해서 사용하는 포트

7-4. 라우팅/라우티드 프로토콜



1. 라우팅 프로토콜 (Routing Protocol)

- 라우터에 살면서 라우티드 프로토콜들에게 목적지까지 가장 좋은 길을 갈 수 있게 해주는 역할
- ex) 자동차를 안전하고 빠르게 운전하는 운전기사

2. 라우티드 프로토콜 (Routed Protocol)

- TCP/IP, IPX, AppleTalk 등 우리가 아는 모든 프로토콜
- 라우팅을 당하는, 라우터가 라우팅 해주는 고객을 뜻함
- ex) TCP/IP나 IPX는 고객으로 라우터라는 자동차를 타고 다른 네트워크로 여행

3. 성능

- 라우터 입장에서 어떤 라우팅 프로토콜(운전사)을 선택하느냐에 따라, 라우터의 성능(얼마나 빨리, 안전하게 가는가)이 결정됨
- 라우터(자동차)가 가지고 있는 기본적인 성능도 중요함

7-4. 라우팅/라우티드 프로토콜



라우티드 프로토콜 : 자동차에 타는 승객

라우팅 프로토콜 : 자동차를 운전하는 것

라우터 : 자동차

라우팅 테이블 : 목적지에 대한 이정표 (운전자마자 모두 다름)

7-5. 스택/다이나믹 라우팅 프로토콜



1. 스택 라우팅 프로토콜

- 한 번 정해놓으면 정해진 그대로 수행하는 프로토콜
- 라우터에 사람이 일일이 경로를 입력해주는 것
- 장점) 사람이 가장 바르고 좋은 길을 찾아서 넣어주므로, 최적의 길임.
라우터는 데이터를 넘기기만 하면 되므로, 생각할 것도 별로 없으며
시키는 대로 하기 때문에 따로 이정표를 많이 가지고 있을 필요도 없음.
라우터 자체에 부담이 들지 않아 라우팅 하는 속도 빨라지고, 성능 좋아짐.
메모리도 적게 들며, 라우터들끼리 라우팅 테이블을 교환할 필요가 없음.
-> 네트워크의 대역폭을 절약할 수 있음.
보안에 강함
- 단점) 사람이 입력해줘야 하므로 귀찮음
입력한 경로에 문제 생길 경우 큰일 발생

7-5. 스테틱/다이나믹 라우팅 프로토콜



2. 다이나믹 라우팅 프로토콜

- 상황에 따라서 그때그때 변화가 가능한 프로토콜
- 장점) 그때그때 가장 좋은 경로를 찾아내는 방식
사람이 일일이 경로 입력 해줄 필요 없음
- 단점) 라우터에 부담을 줌 (라우터가 할 일이 많아지기 때문)
라우팅 프로토콜을 이용해서 어떤 길이 가장 빠른 길인지 계산을 해야함
시간이 지날 때마다 바뀐 정보는 없는지 계속 확인, 이정표 업데이트
- RIP, IGRP, OSPF, EIGRP등이 있음

7-6. 라우팅 테이블



1. 라우팅 테이블

- 라우터가 어떤 경로를 찾을 때 사용하는 것
- 사용하는 라우터의 프로토콜에 따라 달라짐
- 라우터는 항상 최적의 경로를 찾아 라우팅 테이블에 유지
- 전원을 켜는 순간(라우팅 프로토콜에 대한 세팅은 이미 되어 있다고 가정), 라우팅 테이블을 만들기 시작해서 어떤 패킷이 길 안내를 요청하면 라우팅 테이블을 보고 길을 안내함.
- 전원 꺼질 경우 -> 모두 지워짐 -> 전원이 켜지면 다시 만들어짐
- 라우팅 테이블을 시스코 라우터에서 보는 명령어 : `show ip route`

2. 라우팅 테이블의 의미

- I는 IGRP, R은 RIP, O는 OSFF, C는 connected를 나타냄

7-7. AS 그리고 내부용/외부용 라우팅 프로토콜



1. AS(Autonomous System)

- 하나의 네트워크 관리자에 의해서 관리되는 라우터들의 집단
- 하나의 관리 규정 아래서 운용되거나 하나의 관리 전략으로 구성된 라우터 집단
- 한 회사나 기업, 단체의 라우터 집단

2. AS로 묶어주는 이유

- 라우터가 가지는 정보를 효율적으로 관리하고 인터넷 서비스를 좀 더 간편하게 하기 위함
- AS안에 있는 라우터들은 자신의 AS에 속해 있는 라우터에 대한 정보만 알고 있으면 됨
- 외부로 나갈때 AS에 있는 문지기 라우터 (ASBR-Autonomous System Boundary Router)에게 정보를 물어봐서 밖으로(인터넷) 나가는 것.

3. 문지기 라우터

- 자신의 AS와 인접해 있는 다른 AS에 대한 정보를 가지고 있으면서, 자기 AS에서 밖으로 나가는 라우터나 외부 AS에서 자기 AS쪽으로 들어오는 라우터에게 정보 제공
- > 라우터들은 인터넷에 접속하더라도 전 세계 모든 네트워크 정보 X, AS에 대한 정보만 O

7-7. AS 그리고 내부용/외부용 라우팅 프로토콜



4. 프로토콜

- 라우터가 AS내부에서 사용하는 라우팅 프로토콜
 - Interior Routing Protocol 또는 Interior Gateway Protocol(IGP)
- AS외부에서 서로 라우팅 정보를 주고 받기 위해 라우터가 사용하는 프로토콜
 - Exterior Routing Protocol 또는 Exterior Gateway Protocol(EGP)
- 본사와 지사 간 라우터 설치 -> Interior Routing Protocol
- 본사의 라우터가 인터넷 서비스 업체와 연결을 해서 인터넷을 사용 -> Exterior Routing Protocol

7-14. 스테틱 라우팅을 이용한 라우터



- 스테틱 라우팅 프로토콜
 - 스테틱 라우팅 프로토콜은 갈 수 있는 경로가 하나밖에 없는 stub 라우터용으로 많이 사용
- Stub 네트워크
 - 오직 하나의 경로만을 통해서 외부 망과 연결된 네트워크 의미
 - ex) 게임방 네트워크. 대부분 라우터 한 대를 가지고 인터넷에 연결

7-15. 디폴트 라우트



1. 디폴트 라우트

- 경로를 찾아내지 못한 모든 네트워크들은 모두 이곳으로 가라고 미리 정해놓은 길
- 여러가지 라우팅 프로토콜에 의해서 길을 다 찾아간 다음에 그래도 길을 못 찾은 네트워크가 있다면 무조건 이 곳으로 가라고 해줌.

2. 사용

- 인터넷을 사용하는 라우터 (가장 많이 사용하는 인터페이스 쪽으로 디폴트 라우트를 잡아놓으면 다른 경로에서 해당 네트워크를 못 찾을 때는 무조건 인터넷 쪽 인터페이스로 가보게 됨)

- Stub네트워크에 있는 라우터 (스텝 네트워크에 있는 라우터는 갈 수 있는 경로가 하나 밖에 없으니, 그 하나의 길만 디폴트 라우트로 만들어 놓으면 계속 그 길로만 다니게 됨)

(스태틱 라우팅은 특정 목적지를 가기 위한 구성, 디폴트 라우팅은 특정 목적지를 지정하지 않고 모든 목적지가 모두 디폴트로 지정한 곳으로 간다는 것에 차이)

3. 만드는 방법

- 1) 디폴트 네트워크 이용
- 2) 스태틱 명령 이용

7-17. 디스턴스 벡터와 링크 스테이트

1. 디스턴스 벡터 알고리즘

- 디스턴스와 벡터만을 위주로 만들어진 라우팅 알고리즘
 - 라우터는 목적지까지의 모든 경로를 자신의 라우팅 테이블 안에 저장하는 것이 아니라 목적지까지의 거리와 그 목적지까지 가려면 어떤 인접 라우터를 거쳐서 가야 하는 (방향)만을 저장함
 - 인접 라우터들과 주기적으로 라우팅 테이블을 교환하여 자신의 정보에 변화가 생기지 않았는지 확인 및 관리
 - 한 라우터가 모든 라우팅 정보를 가지고 있을 필요가 없기 때문에 라우팅 테이블을 줄일 수 있어서 메모리를 절약하고, 라우팅의 구성 자체가 간단함
 - 라우팅 테이블에 아무런 변화가 없더라도 정해진 시간마다 한 번씩 꼭 라우팅 테이블의 업데이트가 일어남 -> 트래픽 쓸데없이 낭비, 라우팅 테이블에 변화가 생길 경우 이 변화를 모든 라우터가 알 때까지 걸리는 시간 너무 느림
- 단점 때문에, 커다란 네트워크에서는 적용 X, 작은 규모에 적용하면, 구성의 편리와 메모리의 절약

7-17. 디스토크스 벡터와 링크 스테이트



2. 링크 스테이트 알고리즘

- 한 라우터에서 목적지까지의 모든 경로를 알고 있기 때문에 중간에 링크의 변화가 생겨도 이를 알아내는데 걸리는 시간 짧음.
- 라우팅 테이블의 교환이 자주 발생하지 않고, 교환 일어나는 경우에도 테이블의 변화가 있는 것만을 교환하므로 트래픽 발생을 줄여줄 수 있음
- 라우터가 모든 라우팅 정보를 관리해야 하므로, 메모리를 많이 소모
- SPF 계산 등 여러가지 계산을 해야 하기 때문에 라우터 CPU가 일을 많이 해야함
→ 커다란 네트워크에 설치되는 고용량 라우터에 적용하는 것 바람직

7-19. 시스코 라우터의 친구 찾기 CDP



1. CDP

- 시스코 장비를 찾아내는 프로토콜
- 시스코 장비와 직접 연결되어 있는 시스코 장비만을 찾아냄
- 타사 장비 라우터 제외, 해당 라우터와 직접 연결되어 있지 않고 스위치를 통해 연결된 경우도 제외
- Data Link 계층에 올라가는 프로토콜 → 상위 네트워크 계층에 어떤 프로토콜이 올라가는가에 상관없이 실행 가능 (CDP가 데이터 링크 계층에 있기 때문에, IP주소 세팅 X)
- 현재 시스코 장비에 접속된 상대 장비의 IP주소나 기타 정보 모를 때, 어떤 포트로 접속되었는지 알고 싶을 때 사용하면 편리
- CDP는 멀티캐스트를 이용해서 시스코 장비들을 찾아냄

7-21. 핑과 트레이스



- 핑과 트레이스
 - 라우터를 구성한 후 네트워크의 연결에 이상이 없는지를 테스트하기 위해 만듦
 - 출발지에서 목적지까지 연결에 이상이 없는지, 이상이 있다면 어디에서 발생했는지 핑과 트레이스를 이용하여 찾아낼 수 있음.
- 단순형 핑을 사용했을 때 출발지 주소는 라우터를 떠나는 쪽 인터페이스로 자동으로 잡히기 때문에 출발지 주소를 변경해 주려면 반드시 확장형 핑을 사용
- 트레이스 (목적지까지의 경로를 하나하나 분석해주는 기능) : TTL이라는 값을 하나씩 증가시키면서 돌아오는 에러 메시지를 가지고 경로를 확인해주는 기능 제공

8-1. RIP라는 라우팅



1. 다이내믹 라우팅 프로토콜

- 운영자가 일일이 경로를 지정하지 않아도 라우터가 알아서 길을 찾아가는 프로토콜
- 다이내믹 라우팅 프로토콜 중 쉽고 단순한 RIP

2. RIP(Routing Information Protocol)

- 라우팅 프로토콜
- 다이내믹 프로토콜
- 내부용 라우팅 프로토콜 (IGP)
- 디스턴스 벡터 알고리즘 (거리와 방향으로 길을 찾아가는 프로토콜)
- 좋은 길을 결정하는 기준 → 홉 카운트
- RIP 라우팅 프로토콜에서 최대한 갈 수 있는 홉 카운트의 거리 → 15개까지
- RIP의 디폴트 라우팅 업데이트 주기 → 30초

3. 단점

- 경로 선택을 오로지 홉 카운트(목적지 가는데 거치는 네트워크 수)에 의존하며, 데이터를 최대 라우팅할 수 있는 거리가 15로 짧다 → 소규모 네트워크

8-3. Distance-Vector 라우팅



1. Distance-vector의 문제점

1) 시간

- 한 번 배운 라우팅 테이블을 계속 전달 → 업데이트가 모든 네트워크에 전달되는 시간이 많이 걸림 (→ 루핑이 발생할 수 있음)
- 라우팅에 루핑이 발생하는 이유 → 한 라우터가 라우팅 정보에 대한 모든 정보를 가지고 있지 못하고, 이웃 라우터로부터의 업데이트가 느리게 이루어지기 때문.

2. 해결방안

1) Maximum Hop Count

- RIP라우팅 프로토콜의 경우는 최대 홉카운트를 15로 규정하면, 라우팅의 루핑이 발생하더라도 16까지 이르게 되면 멈춰줄 수가 있음

2) Hold down Timer

- hold down 타이머가 동작하고 있는 동안에는 외부에서 해당 네트워크에 대한 라우팅 경로 정보를 받았을 때, 원래 가지고 있던 홉카운트보다 큰 값이 들어오면 무조건 무시, hold down 카운터가 종료되거나 목적지에 대한 새로운 경로가 지금 가지고 있던 메트릭과 같거나 좋은 경로가 들어올 때만 이웃 라우터로부터 업데이트를 받아 들임.

8-3. Distance-Vector 라우팅



3) 스플릿 호라이즌

- 라우팅 정보가 들어온 곳으로는 같은 정보를 내보낼 수 없다는 것
- 만약, 하나의 라우터가 어느 네트워크 정보를 인접한 라우터에게 받았다면, 그 인접한 라우터가 그 네트워크에 더 가까이 있을 것이 분명하므로 다른 라우터들로부터 더 이상 받을 필요 없다는 뜻.
- 두 라우터 간의 루핑만을 막기 위해 만들어진 기술 → 전체 라우터 네트워크의 루핑을 스플릿 호라이즌으로 막기 어려움.

4) 라우트 포이즈닝

- 포이즈닝 : 라우팅 테이블에 극악 처방을 한다는 것
- 네트워크 다운시 라우터가 네트워크에 대한 메트릭 값을 사용할 수 없는 값으로.
- 라우팅 테이블에서 지우지는 않음
- 다운된 네트워크를 먼저 무한대치로 바꾸어 버리는 방식
- 잘못된 라우팅 정보를 받는 일을 미리 막을 수 있는 효과

5) 포이즈즌 리버스

- 스플릿 호라이즌처럼 라우팅 정보를 보내온 쪽으로 알려주지 않는 것이 아니라, 라우팅 정보를 되돌려 보내기는 하되 이 값을 무한대 값으로 쓰는 방식

8-4. IGRP 라우팅 프로토콜



1. IGRP

- 디스턴스 벡터 라우팅 프로토콜 중 하나
- 라우팅 프로토콜
- 다이내믹 프로토콜
- 내부용 라우팅 프로토콜
- 디스턴스 벡터 알고리즘
- IGRP 라우팅 프로토콜은 RIP와 달리 표준 프로토콜이 아닌 시스코에서 만들어낸 프로토콜이므로, 시스코 라우터에서만 사용이 가능함
- 90초에 한 번씩 라우팅 테이블의 업데이트
- 최대 홉 카운트는 100이지만, 255까지 조정 가능

2. 경로 선택 방법

- 1) Bandwidth (대역폭) : 속도
- 2) Delay (지연)
- 3) Reliability (신뢰성)
- 4) Load (부하, 하중)
- 5) MTU(경로의 최대 전송 유닛의 크기)

8-5. OSPF 라우팅 프로토콜



1. OSPF(Open Shortest Path First)

- IP 패킷 안에 프로토콜 넘버 89로 들어가게 됨.

2. 특징

1) Convergence Time

- 라우터 간에 서로 변경된 정보를 주고받는데 걸리는 시간
- 어떤 인터페이스가 죽었을 때 이 정보가 모든 라우터들 사이에 퍼지는데 걸리는 시간
- RIP의 경우 매30초에 한 번씩 업데이트가 일어나고, 컨버전스에 많은 시간이 걸리지만 **OSPF는 어떤 변화가 생길 때 바로 전달이 가능하기 때문에 훨씬 빠름!**
- 큰 네트워크에 적당. 빠른 업데이트 하면서 효율적인 관리 가능

2) VLSM 지원 여부

- 각 서브넷마다 가변 길이의 서브넷 마스크를 적용하는 기법
- RIP v1, IGRP - 지원X / OSPF, RIP v2 - 지원 O
- IP주소를 효과적으로 사용할 수 있으며, 라우팅 테이블을 줄이는 효과
- 이를 위해 라우트 서머리제이션을 지원하기 때문에 여러 개의 라우팅 경로를 하나로 묶어주는 기능 탁월

8-5. OSPF 라우팅 프로토콜



3) 네트워크 크기에 대한 제한

- RIP의 경우, 최대 15개의 홉 카운트만 가능하지만, OSPF는 이런 제한이 없음

4) 네트워크 대역폭의 활용

- RIP의 경우 매 30초마다 브로드캐스트가 발생하기 때문에 대역폭 낭비가 많지만, OSPF는 네트워크 변화가 있을 때만 정보가 멀티 캐스트로 날아가기 때문에 실용적

5) 경로 결정

- RIP는 홉 카운트만을 따지기 때문에 속도와 딜레이와 상관없이 홉 카운트가 적은 것만을 선호하지만, OSPF는 많은 관련 요소를 합쳐 경로를 선택 → 정확한 경로 선택 가능

6) 표준 라우팅 프로토콜

7) 링크스테이트 라우팅 알고리즘

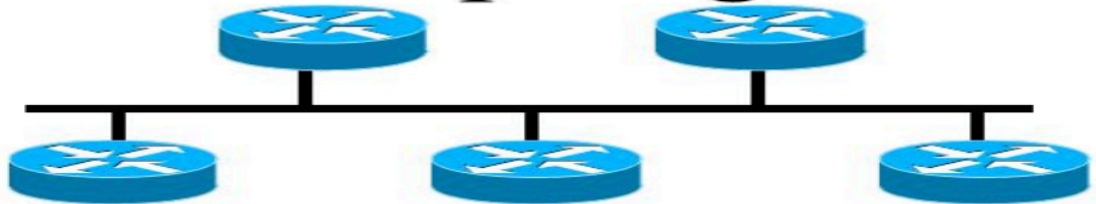
8-5. OSPF 라우팅 프로토콜

8) 토폴로지

- 이더넷 세그먼트
- 한 쌍의 라우터 (전용선)
- 두 개 이상의 라우터 연결되나, 브로드 캐스트 능력 없음

OSPF Topologies

**Broadcast
Multiaccess**



Point-to-Point



NBMA



8-5. OSPF 라우팅 프로토콜



3. 다른 OSPF 라우터와의 교류

- OSPF에서 라우터는 주위에 있는 OSPF 라우터들을 찾아서 자신의 데이터베이스 안에 저장하는데, 이런 주위의 라우터들을 Neighbor(이웃)라고 함.
- 주변에 어떤 이웃들이 사는지에 대한 정보를 관리하는 것
- 이렇게 이웃을 찾아내기 위해서 OSPF 라우터는 Hello 패킷을 내보냄

<Hello 패킷 구성>

A - B 라는 라우터가 인접해 있다고 가정

- 1) A 라우터가 켜지면 그 라우터는 이웃이 누구인지 모르기 때문에 모든 OSPF 라우터들에게 헬로 패킷을 보내는데, 브로드캐스트가 아닌 **멀티캐스트** 주소 224.0.0.5를 이용해서 헬로 패킷을 보낸다. (224.0.0.5라는 멀티캐스트 주소로 보내진 헬로 패킷을 OSPF로 동작하는 라우터만 수신한다)
 - 2) 헬로 패킷을 받은 다른 라우터들(B 라우터)은 라우터 A를 자신들의 이웃목록(Neighbor list)에 넣게 되고, 이 과정을 Init 과정이라 한다.
 - 3) A에게서 헬로 패킷을 받은 OSPF 라우터들이 A에게 **유니캐스트**로 자신들의 정보를 보내게 된다.
 - 4) 라우터 A는 Neighbor들로부터 받은 정보를 자신의 Neighbor 리스트에 넣어 관리한다.
- * Neighbor 관계가 형성이 되어야 통신 가능

8-5. OSPF 라우팅 프로토콜



4. OSPF에서의 반장과 부반장

- DR(Designated Router)과 BDR(Backup Designated Router)
- OSPF 세그먼트에서는 각 라우터들이 OSPF에 참여 → DR과 BDR에게 자신의 Link State를 알림

ex) OSPF라는 반에 전학 온 학생이 반장(DR)과 부반장(BDR)에게 링크 상태 보고

- OSPF에서는 모든 라우터가 반드시 DR, BDR과 Link state 일치
- DR과 BDR은 라우터 ID와 라우터의 Priority로 선출(default = 1)

* ex) DR(Priority = 2)과 BDR(Priority = 1)의 선출 끝난 후, Priority = 3인 라우터 등장
→ 이 경우에는 선출이 모두 끝난 상태이므로, Priority = 3인 라우터는 DR이 될 수 없다.

DR이 다운 되면 BDR이 DR이 되고, Priority = 3인 라우터가 BDR의 자리

→ 만약 라우터들을 전부 꺾다가 다시 키거나, OSPF를 전부 죽였다가 살리면,
Priority = 3인 라우터가 DR로 선출될 수 있다.

- DR, BDR의 권한을 주기 싫을 때 → 라우터의 priority 값을 0으로 준다.

- Priority 값이 같은 경우 → Router ID를 비교

9-8. IP주소의 변환 NAT



- NAT(Network Address Translation)
 - 한쪽 네트워크의 IP주소가 다른 네트워크로 넘어갈 때, 변환되어 넘어가는 것
- 많이 사용되는 이유
 - 1) 내부의 네트워크에는 비공인 IP주소를 사용하고, 외부 인터넷으로 나가는 경우에만 공인 IP주소를 사용하고자 하는 경우
 - 내부 pc에 부여할 공인 주소 한정되어 있을 경우
 - 2) 기존에 사용하던 ISP에서 새로운 ISP로 바꾸면서 내부 전체의 Ip를 바꾸지 않고 기존의 IP주소를 그대로 사용하고자 하는 경우
 - 내부의 주소를 자주 바꾸고 싶지 않은 경우.
 - 3) 두 개의 인트라넷을 서로 합하려다 보니 두 네트워크의 IP가 겹치는 경우
 - 서로 비공인 주소를 사용하던 두 네트워크를 연결하는 경우 사용하던 IP주소 영역 겹칠 수 있게 됨 -> NAT 사용하면, 두 네트워크 주소 일일이 변경 하지 않고 해결 가능
 - 4) TCP 로드 분배가 필요한 경우
 - 밖에서는 하나의 주소로 보이는 호스트가 내부에서는 여러 개의 호스트에 매핑되도록 하여 서버의 로드를 분배하는 기술