



2.1 네트워크의 기초

🕒 작성일시	@2024년 1월 29일 오후 6:43
☑ 복습	<input type="checkbox"/>

- 네트워크 = 노드와 링크가 서로 연결되어 있거나 연결되어 있지 않은 결합체
 - 노드 = 서버, 라우터, 스위치 등 네트워크 장치
 - 링크 = 유선 또는 무선

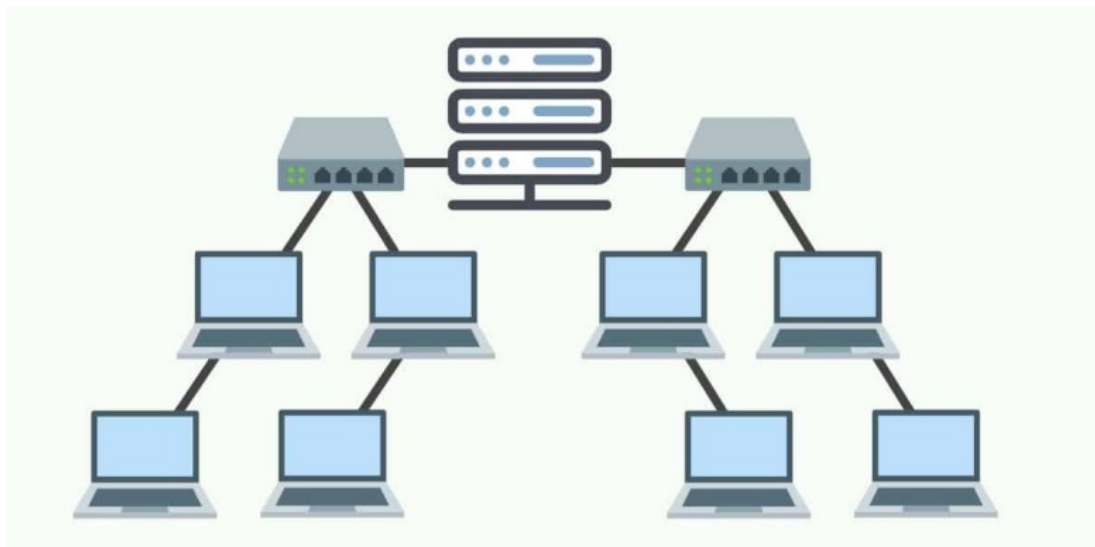
처리량과 지연시간

- 좋은 네트워크 = 많은 처리량을 처리할 수 있으며 지연시간 짧고 장애빈도 적으며 좋은 보안 갖춘 네트워크
- 처리량
 - 링크를 통해 전달되는 단위 시간당 데이터양
 - 단위 : bps(bits per second)
 - 처리량에 영향을 미치는 요소
 - 트래픽 (서버와 스위치 등 네트워크 장치에서 일정 시간 내에 흐르는 데이터의 양)
 - 트래픽이 많다 → 접속한 사람이 많다 → 처리해야할 데이터 양이 많다
 - 대역폭(주어진 시간 동안 네트워크 연결을 통해 흐를 수 있는 최대 비트 수)
 - 대역폭이 크다 → 한번에 많은 데이터 전송 가능 (처리량 ↑)
 - 대역폭이 크다고 속도가 빠른건 아님!
 - 처리량은 대역폭 초과할 수는 x
 - 네트워크 중간에 발생하는 에러
 - 장치의 하드웨어 스펙

- 지연시간
 - 요청이 처리되는 시간
 - 어떤 메시지가 두 장치 사이를 왕복하는데 걸리는 시간
 - 지연시간에 영향을 미치는 요소
 - 매체 타입(유선, 무선)
 - 패킷 크기
 - 라우터의 패킷 처리 시간

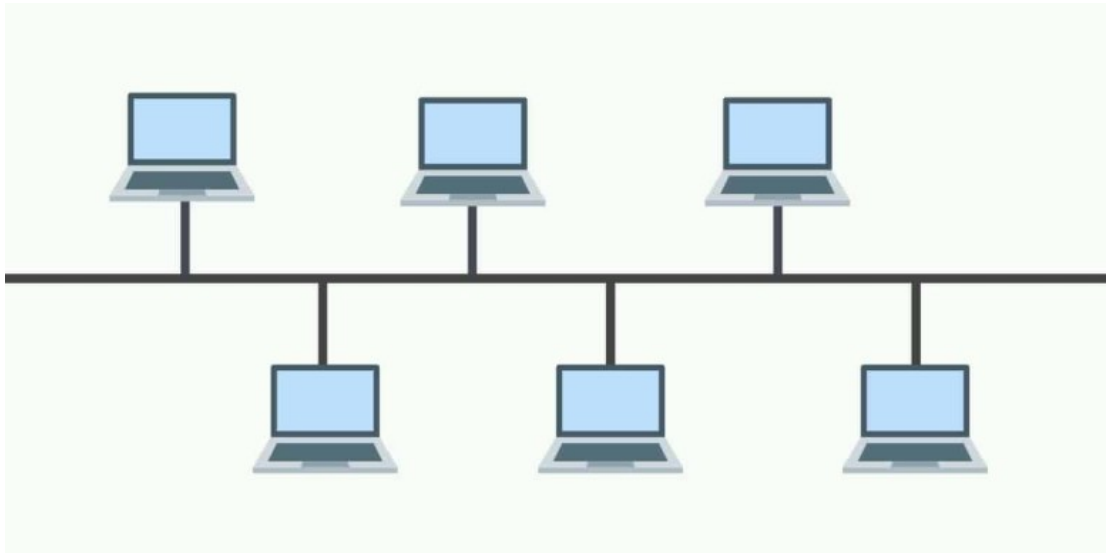
네트워크 토폴로지와 병목현상

- 네트워크 토폴로지 = 노드와 링크가 어떻게 배치되어 있는지에 대한 방식, 연결 형태
- 토폴로지의 종류
 1. 트리 토폴로지 (계층형 토폴로지)



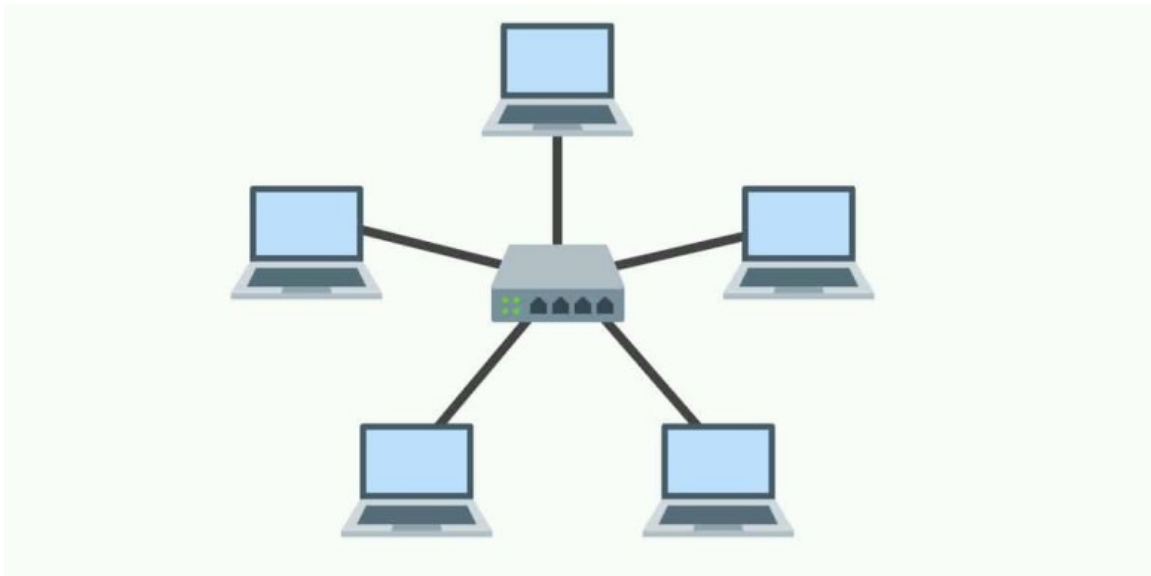
- 계층적으로 연결된 네트워크 구성
- 장점
 - 네트워크 관리가 쉬움(노드의 추가, 삭제가 쉬움)
- 단점
 - 특정 노드에 트래픽이 집중될 때 하위 노드에 영향을 끼칠 수 있음
 - 병목현상이 발생할 수 있음

2. 버스 토폴로지



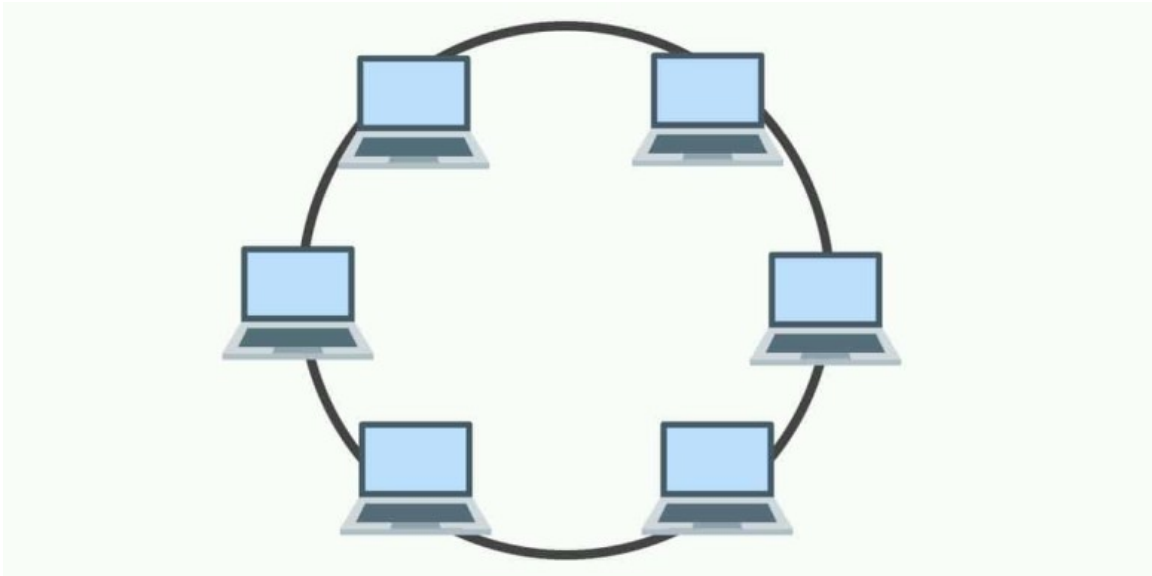
- 중앙 통신 회선 하나에 여러 개의 노드 연결
- 근거리 통신망(LAN)에서 사용
- 장점
 - 설치 비용 적고 신뢰성 우수
 - 중앙 통신 회선에 노드 추가 삭제 쉬움
 - 구조 간단
- 단점
 - 스푸핑이 가능한 문제점
 - 전송되는 데이터가 많으면 병목현상 발생
- 스푸핑
 - 직접적으로 시스템에 침입을 시도하지 않고 피해자가 공격자의 악의적인 시도에 의한 잘못된 정보, 혹은 연결을 신뢰하게끔 만드는 일련의 기법들

3. 스타 토폴로지



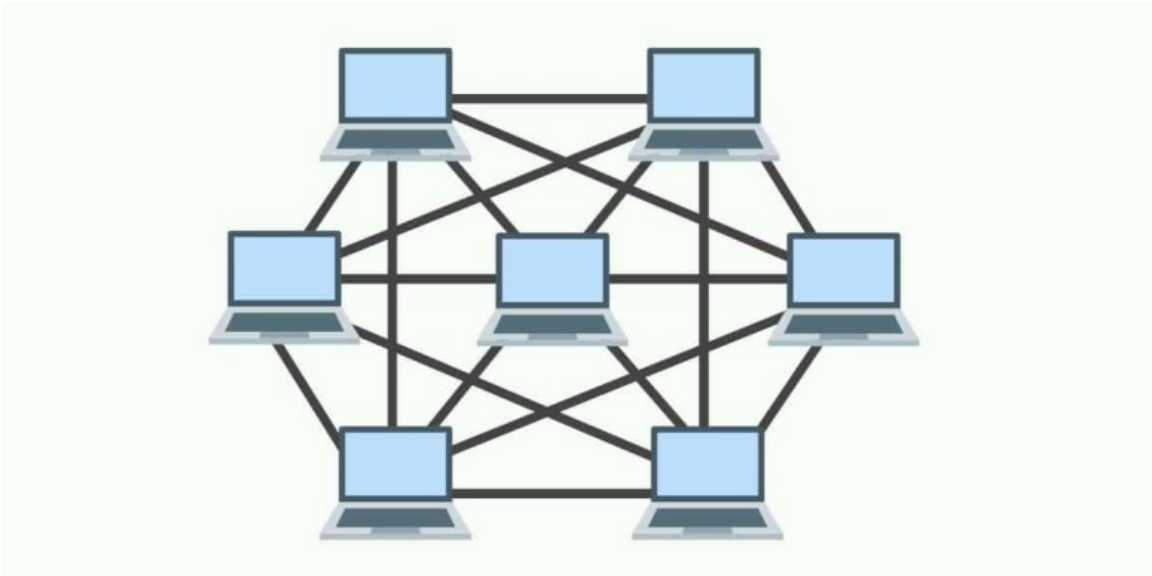
- 중앙에 있는 노드에 모두 연결된 네트워크 구성
- 장점
 - 노드 추가 쉬움
 - 에러 탐지 쉬움
 - 패킷의 충돌 발생 가능성이 적음
 - 노드에 장애가 발생해도 네트워크는 사용 가능(중앙 노드에 발생한 장애가 아니라면)
- 단점
 - 중앙 노드에 장애가 발생하면 전체 네트워크 사용 불가능
 - 설치 비용 고가

4. 링형 토폴로지



- 각각의 노드가 양 옆의 두 노드와 연결하여 전체적으로 링처럼 하나된 연속된 길을 통해 통신
- 데이터는 노드에서 노드로 이동
- 장점
 - 노드 수가 증가 되어도 네트워크 상의 손실이 거의 없음
 - 충돌 발생 가능성 적음
 - 노드 고장 발견 쉽게 찾을 수 있음
- 단점
 - 네트워크 구성 변경이 어려움
 - 회선에 장애가 발생하면 전체 네트워크에 영향

4. 메시 토폴로지



- 모든 노드들이 연결
- 장점
 - 한 단말 장치에 장애가 발생해도 여러개의 경로가 존재해 네트워크 이용가능
 - 트래픽 분산처리 가능
- 단점
 - 구축, 운영 비용 고가
 - 노드 추가 어렵다

병목현상

- 전체 시스템의 성능이나 용량이 하나의 구성요소로 인해 제한을 받는 현상
- 네트워크가 어떤 토폴로지를 갖는지 알아야 병목현상 올바르게 해결 가능

네트워크 분류

- LAN(Local Area Network)
 - 근거리 통신망
 - 집, 학교, 사무실 등 좁은 공간에서 운영
 - 전송속도 빠르고 혼잡하지 않음

- MAN(Metropolitan Area Network)
 - 대도시 지역 네트워크
 - 도시같은 넓은 지역
 - 전송속도 평균, LAN보다는 혼잡함
- WAN(Wide Area Network)
 - 원거리 통신망
 - 국가 대륙등 넓은 지역 연결
 - 전송속도 낮으며 혼잡

네트워크 성능 분석 명령어

- 코드 상 문제 없는데 사용자가 서비스로부터 데이터를 가지고 오지 못하는 상황 발생 → 병목현상일 가능성 존재
- 병목현상 주된 원인
 - 네트워크 대역폭
 - 네트워크 토폴로지
 - 서버 CPU, 메모리 사용량
 - 비효율적인 네트워크 구성

⇒ 네트워크로부터 발생한 문제점인 것 확인한 후 네트워크 성능 분석

- ping(Packet INternet Groper)
 - 네트워크 상태 확인하려는 대상노드 향해 일정 크기의 패킷을 전송
 - 이를 통해서 패킷 수신 상태와 도달하기까지 시간 등 알 수 있음
 - 해당 노드까지 네트워크가 잘 연결되어 있는지 확인 가능
 - TCP/IP 프로토콜 중 ICMP 프로토콜(사용자 데이터의 전송 과정에서 오류가 발생 하면 오류 메시지 전송해주는 기능을 담당하는 프로토콜)을 통해 동작
 - **ping[IP 주소 또는 도메인 주소]**
 - ping 옵션

-t	중지될 때까지 지정한 호스트를 ping함
----	------------------------

-n	지정한 숫자만큼 패킷 보냄
-a	주소를 호스트 이름으로 확인

```
C:\Users\user>ping www.google.com -n 5

Ping www.google.com [142.250.206.196] 32바이트 데이터 사용 :
142.250.206.196의 응답: 바이트=32 시간=60ms TTL=54
142.250.206.196의 응답: 바이트=32 시간=59ms TTL=54
142.250.206.196의 응답: 바이트=32 시간=72ms TTL=54
142.250.206.196의 응답: 바이트=32 시간=60ms TTL=54
142.250.206.196의 응답: 바이트=32 시간=60ms TTL=54

142.250.206.196에 대한 Ping 통계:
    패킷: 보냄 = 5, 받음 = 5, 손실 = 0 (0% 손실),
    왕복 시간(밀리초):
        최소 = 59ms, 최대 = 72ms, 평균 = 62ms
```

- netstat
 - 접속되어 있는 서비스들의 네트워크 상태를 표시
 - 네트워크 접속, 라우팅 테이블, 네트워크 프로토콜 등을 보여줌
 - 서비스 포트가 열려있는지 확인할 때 사용
 - netstat 상태

TIME_WAIT	이미 해당 사이트와 연결이 종료되었거나 다음 연결을 위해 기다리는 상태
CLOSE_WAIT	원격의 연결 요청을 받고 연결이 종료되길 기다리는 상태
ESTABLISHED	서로 연결되어 있는 상태
LISTEN	연결 요구를 기다리는 상태 포트가 열려있음
CLOSED	완전히 연결이 종료된 상태


```
C:\Users\user>netstat
```

활성 연결

프로토콜	로컬 주소	외부 주소	상태
TCP	127.0.0.1:64702	DESKTOP-HVCQC5S:64703	ESTABLISHED
TCP	127.0.0.1:64703	DESKTOP-HVCQC5S:64702	ESTABLISHED
TCP	127.0.0.1:64704	DESKTOP-HVCQC5S:64705	ESTABLISHED
TCP	127.0.0.1:64705	DESKTOP-HVCQC5S:64704	ESTABLISHED
TCP	127.0.0.1:64706	DESKTOP-HVCQC5S:64707	ESTABLISHED
TCP	127.0.0.1:64707	DESKTOP-HVCQC5S:64706	ESTABLISHED
TCP	127.0.0.1:64708	DESKTOP-HVCQC5S:64709	ESTABLISHED
TCP	127.0.0.1:64709	DESKTOP-HVCQC5S:64708	ESTABLISHED
TCP	127.0.0.1:64710	DESKTOP-HVCQC5S:64711	ESTABLISHED
TCP	127.0.0.1:64711	DESKTOP-HVCQC5S:64710	ESTABLISHED
TCP	127.0.0.1:64714	DESKTOP-HVCQC5S:64715	ESTABLISHED

- nslookup
 - DNS에 관련된 내용을 확인하기 위한 명령어
 - 특정 도메인에 매핑된 ip 확인

```
C:\Users\user>nslookup
기본 서버:  kns.kornet.net
Address:  168.126.63.1

> google.com
서버:      kns.kornet.net
Address:  168.126.63.1

권한 없는 응답:
이름:      google.com
Addresses:  2404:6800:400a:80a::200e
            172.217.25.174
```

- tracert
 - 목적지 노드까지 네트워크 경로를 확인할 때 사용하는 명령어

- 어느 구간에서 응답시간이 느려지는지 확인 가능

```
C:\Users\user>tracert www.google.com

최대 30홉 이상의
www.google.com [142.250.206.196](으)로 가는 경로 추적:

  1    <1 ms    <1 ms    <1 ms    192.168.1.1
  2     3 ms     6 ms     2 ms    192.168.111.1
  3    14 ms    12 ms    11 ms    10.53.0.1
  4    10 ms    12 ms     8 ms    192.168.144.81
  5     9 ms    11 ms     7 ms    192.168.144.133
  6    12 ms    14 ms     7 ms    211.61.160.41
  7    16 ms    13 ms    16 ms    220.64.25.77
  8    19 ms    17 ms    17 ms    211.61.192.5
  9    19 ms    15 ms    16 ms    211.242.42.110
 10   588 ms   862 ms   1095 ms    192.145.251.168
```

- ftp(파일 전송하여 테스트) / tcpdump 명령어 (노드로 오고가는 패킷 캡처) / wireshark(네트워크 패킷 캡처 및 분석 소프트웨어) / netmon(네트워크 패킷 모니터링)

네트워크 프로토콜 표준화

- 네트워크 프로토콜 = 다른 장치들끼리 데이터를 주고받기 위해 설정된 공통된 인터페이스
- IEEE / IETF 표준화 단체가 프로토콜 정함
- HTTP라는 프로토콜 통해 노드들은 웹 서비스 기반으로 데이터 주고받음