

## chapter 5



# 네트워크 운영체제

**NOS** (Network Operating System)

# 5장 내용

---

- 네트워크 운영체제의 공통적 기능
- 네트워크 운영체제의 발전 역사
- 4개의 대표적 네트워크 운영체제 소개
- 여러가지 네트워크 관리 툴들

# 네트워크 운영체제 – 소개

- 네트워크 운영체제는
  - 네트워크 프로그램과 네트워크 하드웨어를 연결해 주는 소프트웨어
  - 서로 다른 프로그램들이 네트워크 기능을 공유하도록 관리해주는 소프트웨어

# NOS – 공통기능

- 현대 NOS의 공통 기능
  - 파일 시스템 서비스
  - 프린팅 서비스
  - 어플리케이션 서비스
  - 웹 파일공유 서비스
  - 웹기반 관리
  - 서버 클러스터링
  - 디렉토리 서비스
  - 보안 서비스
  - 인터넷과 웹서비스

# NOS – 필요성

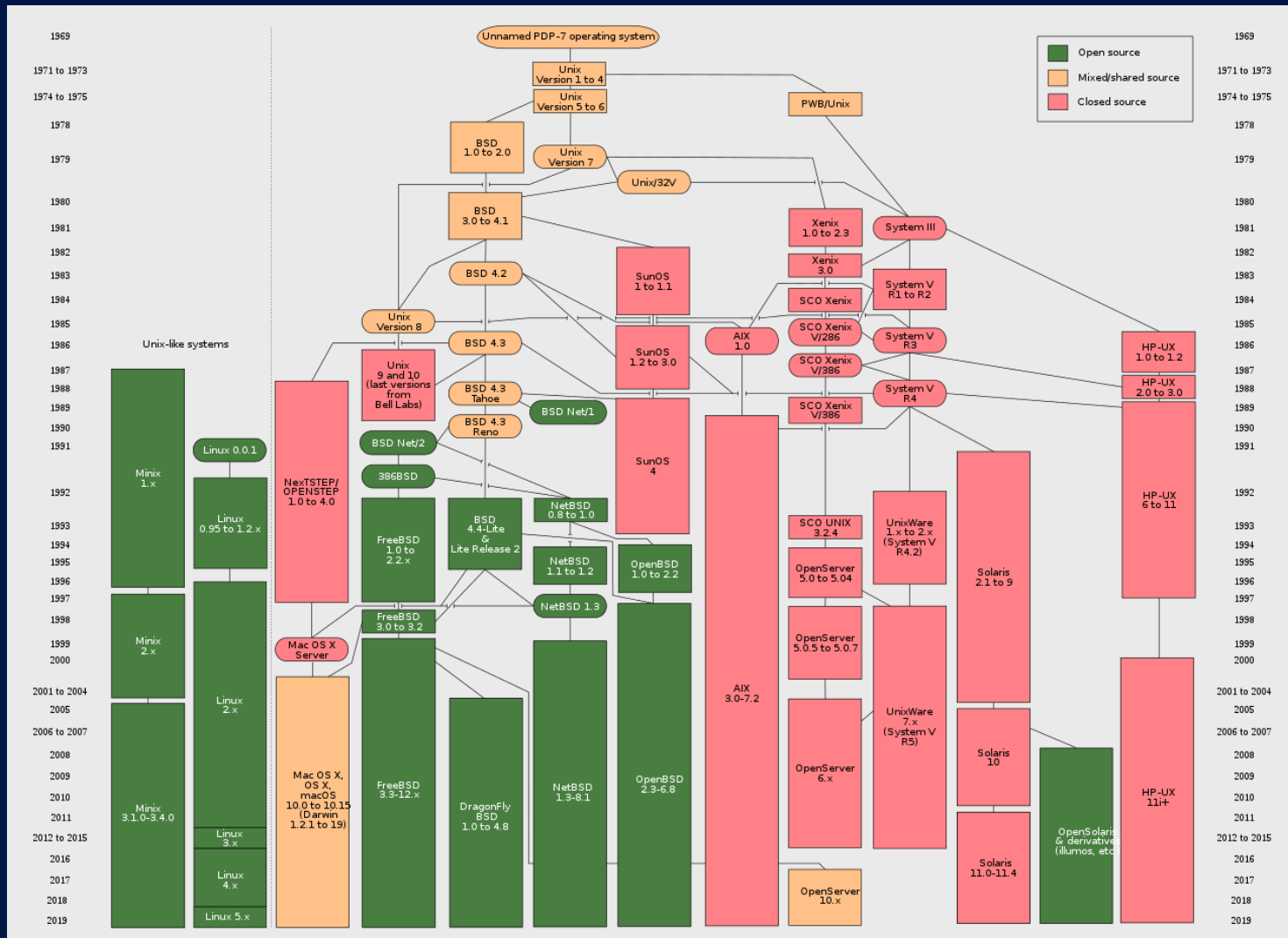
- 최초
  - Application에서 직접 Network 구현
    - HW 호환성 문제 -> Network Service 프로그램으로 변형
      - NIC, 서비스 프로그램, 서비스 프로그램 사용 SDK
  - 보안 문제, 중복 구현 문제, 공유 문제
  - 운영체제 안으로 넣으면서 Kernel Interface 필요
    - TCP/IP는 Socket Programming API
- 용도 변경
  - 정보 교환에서 시작
  - 속도 증가 -> 리소스 교환 -> 리소스 공유 -> 보안 문제
  - 운영체제의 체계적인 관리 필요

# NOS - UNIX

- 1969년 개발시작, 1971년 내부 발표, 1973년 외부 발표
- AT&T 벨연구소에서 켄 톰슨과 데니스 리치가 개발
- 모든 현대 운영체제의 근본
  - 프로세스, 시스템 호출, 파일 관리, 네트워킹(BSD) 등등
  - C로 프로그래밍 하면 운영체제에 관계없이 잘 실행되는 원인
- 수 많은 버전이 있음
  - 컴퓨터 회사마다 라이선스를 구입해서 자기 버전을 만들어 자기 컴퓨터에 설치해서 판매 => 미묘하게 달라서 더 짜증!!!
- TCP/IP는 BSD 4.1c에서 처음 구현되어서 역수입 되었음.
  - MS에서도 가져감
  - BSD(Berkley Software Distribution) 캘리포니아 주립대학 버클리 분교에서 만든 버전

# NOS - UNIX

<https://en.wikipedia.org/wiki/Unix>



# NOS - UNIX

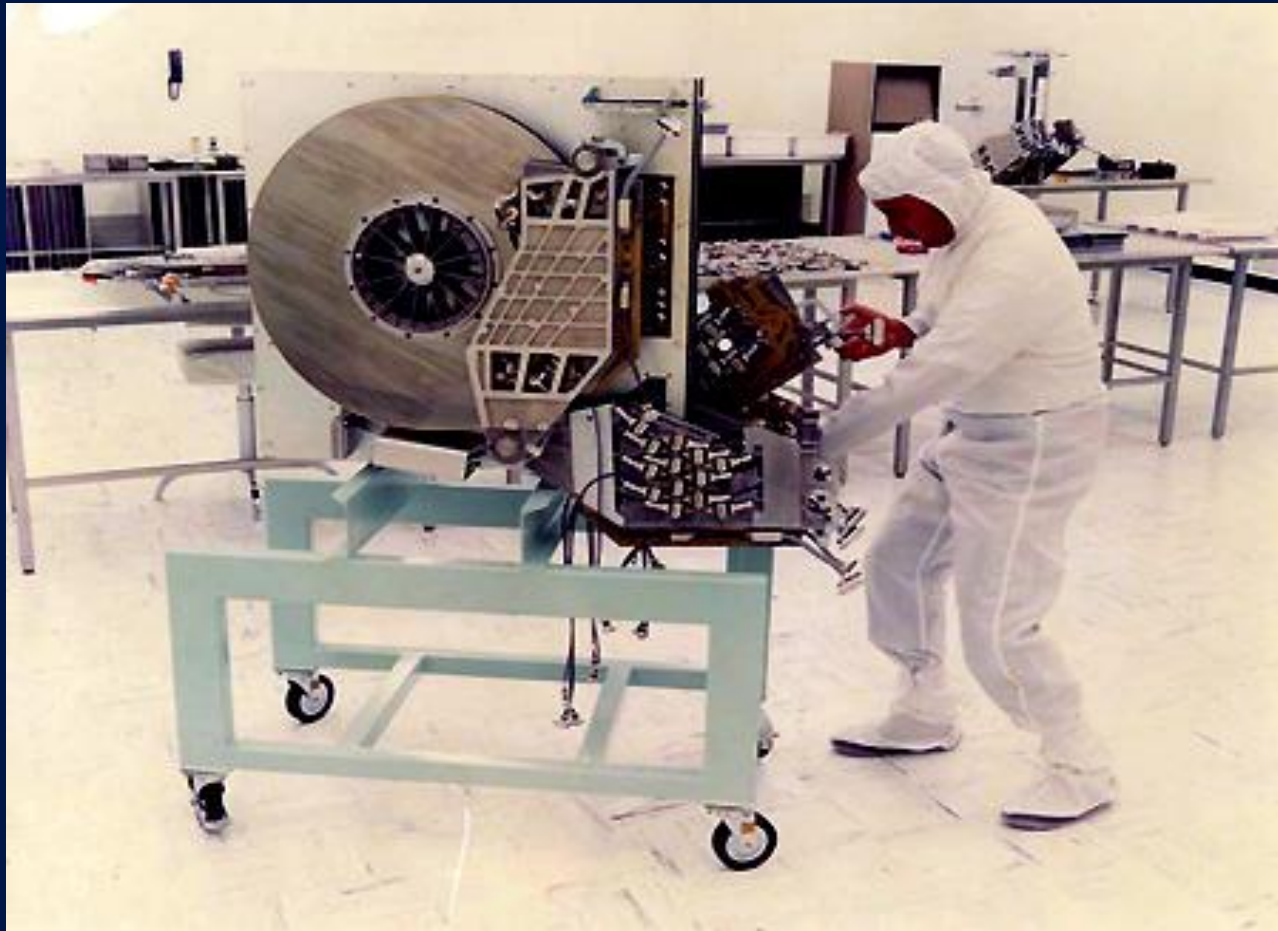
- UNIX와 TCP/IP는 서로 시너지 효과를 내면서 같이 폭발적으로 발전
- Socket Programming을 통한 쉬운 네트워크프로그래밍 SDK제공
  - File I/O와 거의 유사한 프로그래밍 API
- 지금 까지도 많이 사용되는 네트워크 유틸리티 제공
  - IFCONFIG, ARP, NETSTAT, PING, TELNET...



# PC용 NOS – 최근역사

- 사업체를 위한 NOS의 진화
  - 1980년대 초반은 하드디스크를 모든 PC에 따로 장착하기에는 너무 비쌌다.
  - 사용자들의 파일 생성 및 사용 용량이 점점 빨라졌다.
  - PC가 보급되면서, 업체에서는 쉽고 저렴하게 파일을 공유하는 수단이 필요해졌다.
  - PC의 성능이 올라가면서 메인프레임에 접속해서 작업을 하기 보다는 PC에서 자체적으로 처리하는 것이 더 가성비가 좋아졌다.
  - 결국에는 TCP/IP 인터넷에 모두 흡수 당함.

# What is this?



# Is this real?



# PC용 NOS – Novell NetWare

- 1983년도에 첫 버전 출시
- IBM-PC (MS-DOS)에서 네트워크를 통한 파일공유 제공
  - 네트워크를 통한 프린팅 기능 제공
  - 운영체제가 아니라 운영체제 위에 올라가는 추가 서비스
- 기업 사무실 맞춤형 환경 제공
  - 학교 연구실/실습실 포함
- TCP/IP가 아니라 전용 프로토콜인 IPX/SPX프로토콜 사용
  - 망한 첫번째 이유
  - 결국에는 TCP/IP 옮겼지만 기술 차별화에 실패
- NetWare 4.0에서 디렉토리 서비스 제공
  - 디렉토리 서비스 (Directory Service)
  - 컴퓨터, 파일, 서비스를 네트워크에서 찾아주는 기능
- PC 파일/프린터 공유기능의 원조로 한때 정말 많이 사용

# NOS – 마이크로소프트 윈도우즈

- 1993년 Windows NT 3.1이 나오기 전까지는 MS 운영체제에서 인터넷 표준을 지원하지 않았음.
  - MS-NET(1985) : IPX/SPX상에서 돌아가는 IBM의 NetBIOS기반 네트워크
  - LAN Manager(1987) : IPX/SPX위의 NetBIOS를 통한 XNS 프로토콜
  - MSN (1995) : 전화 연결 네트워크 – 최후의 발악.
  - Windows 10에 와서 흑역사 세탁완료.
- 개인용 Windows는 1996년 Windows 95 Service Pack1 부터 인터넷 지원
  - 그 이전까지는 별도의 TCP/IP 관리 프로그램을 설치해야 했음.
- 이후부터는 TCP/IP표준 지원
  - 소켓 프로그래밍 라이브러리에 이전의 흔적이 남아 있음.

# NOS - Linux

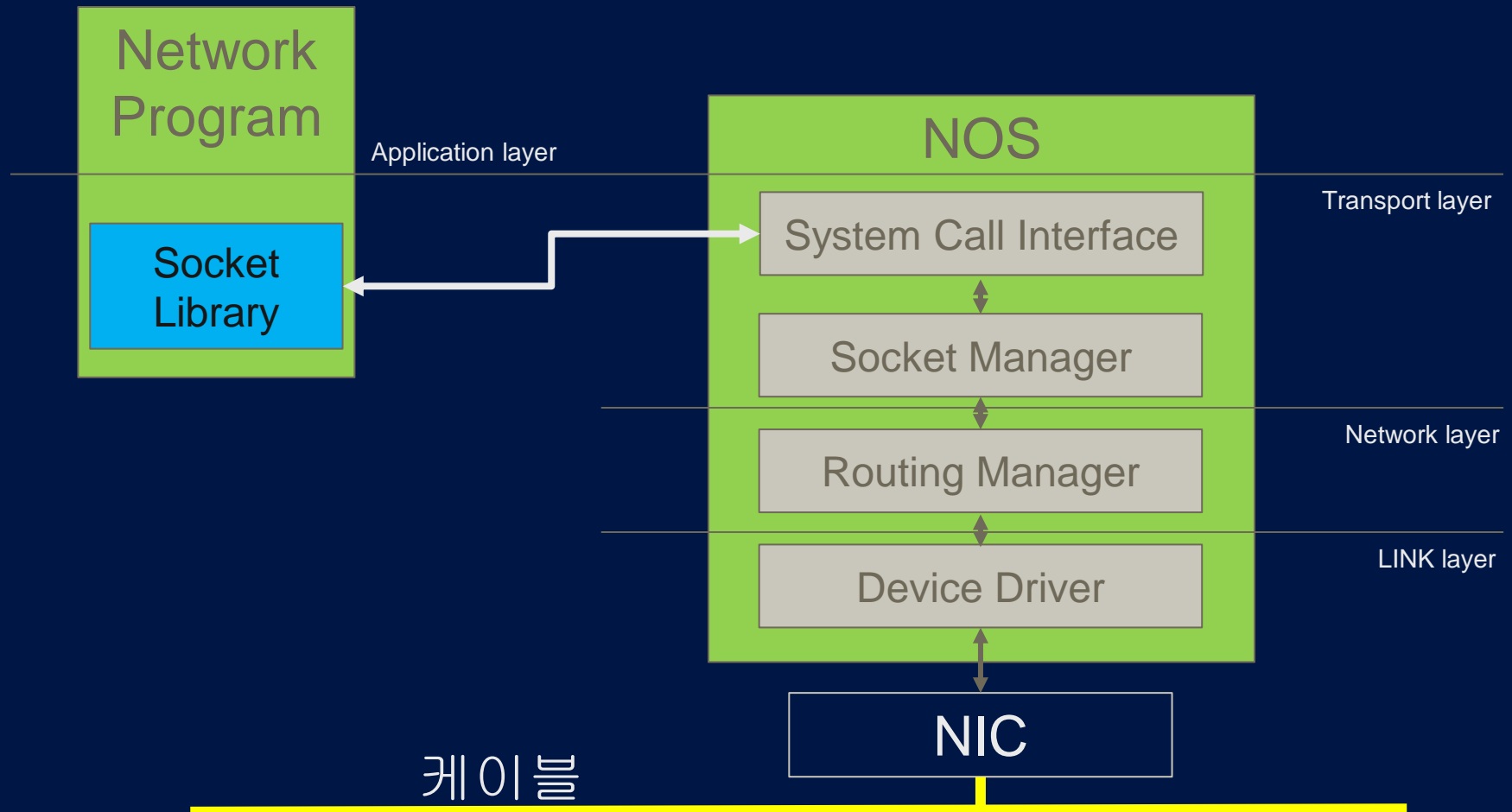
- UNIX에 근원을 둔 운영체제
  - UNIX하고 사용법이 똑 같다.
  - 하지만 내부는 완전히 따로 개발
- 1991년 Linus Torvalds 주도로 개발
- 1994년에 Linux v1.0 발표
- **오픈소스**이다.
- PC에서 출발해 이제는 메인프레임이나 슈퍼컴퓨터에서도 사용. Android도 리눅스
- UNIX의 모든 네트워크 기능을 가지고 있으며 고유의 부가 기능들도 계속 추가 되고 있음.
- 온라인 게임서버 운영체제로 해외에서 많이 사용
  - 우리나라만 특이하게 Windows를 많이 사용하고 있음
  - 점점 LINUX의 비중이 늘어나고 있음.

# NOS - Linux

- 웹서버 운영체제 비율 ( [https://en.wikipedia.org/wiki/Usage\\_share\\_of\\_operating\\_systems](https://en.wikipedia.org/wiki/Usage_share_of_operating_systems) )

| Source         | Date     | <u>Unix, Unix-like</u> |              |                |         | <u>Microsoft Windows</u> | References |
|----------------|----------|------------------------|--------------|----------------|---------|--------------------------|------------|
|                |          | All                    | <u>Linux</u> | <u>FreeBSD</u> | Unknown |                          |            |
| W3Techs        | Nov2020  | 71.8%                  | 40%          | 0.5%           | 31.3%   | 28.2%                    | [227]      |
| W3Techs        | Feb 2015 | 67.8%                  | 35.9%        | 0.95%          | 30.9%   | 32.3%                    | [228][229] |
| Security Space | Feb 2014 | <79.3%                 | N/A          |                |         |                          | [230][231] |

# NOS 구조





# 네트워크 관리 툴

## ● ARP

- 'arp -a' 로컬랜에 연결된 NIC들을 보여줌

```
C:\Users\nhjung>arp -a

인터페이스: 192.168.1.17 --- 0xf
  인터넷 주소      물리적 주소      유형
  192.168.1.1      e8-fc-af-fa-xx-xx  동적
  192.168.1.6      68-ec-c5-99-xx-xx  동적
  192.168.1.14     54-27-1e-e6-xx-xx  동적
  192.168.1.32     34-36-3b-d3-xx-xx  동적

인터페이스: 192.168.0.14 --- 0x17
  인터넷 주소      물리적 주소      유형
  192.168.0.1      00-26-66-a5-xx-xx  동적

C:\Users\nhjung>
```

- 'arp -s' 항목 추가

```
예:
> arp -s 157.55.85.212 00-aa-00-62-c6-09 .... 정적 항목을 추가합니다.
> arp -a .... ARP 표를 표시합니다.
```

# 네트워크 관리 툴

- IPCONFIG
  - NIC에 할당된 IP주소 확인

```
C:\Users\nhjung>ipconfig

Windows IP 구성

이더넷 어댑터 이더넷 5:

    미디어 상태 . . . . . : 미디어 연결 끊김
    연결별 DNS 접미사. . . . :

이더넷 어댑터 이더넷 3:

    연결별 DNS 접미사. . . . :
    링크-로컬 IPv6 주소 . . . . : fe80::e023:3e5a:23dd:2da0%23
    IPv4 주소 . . . . . : 192.168.0.14
    서브넷 마스크 . . . . . : 255.255.255.0
    기본 게이트웨이 . . . . . : 192.168.0.1

무선 LAN 어댑터 Wi-Fi 3:

    연결별 DNS 접미사. . . . :
    링크-로컬 IPv6 주소 . . . . : fe80::5a1:15d8:a9b3:2730%15
    IPv4 주소 . . . . . : 192.168.1.17
    서브넷 마스크 . . . . . : 255.255.255.0
    기본 게이트웨이 . . . . . : 192.168.1.1

C:\Users\nhjung>
```

# 네트워크 관리 툴

## ● IPCONFIG

- ‘ipconfig /renew’ IP주소 다시 받아 오기
  - DHCP 다시 요청하기

```

예제:
> ipconfig                ... 정보 표시
> ipconfig /all           ... 자세한 정보 표시
> ipconfig /renew         ... 모든 어댑터 갱신
> ipconfig /renew EL*     ... EL로 시작되는 이름을 가진
                        ... 모든 연결 갱신
> ipconfig /release *Con* ... 모든 일치하는 연결 해제
                        예: "Wired Ethernet Connection 1" or
                        "Wired Ethernet Connection 2"
> ipconfig /allcompartments ... 모든 컴파트먼트에 대한
                        ... 정보 표시
> ipconfig /allcompartments /all ... 모든 컴파트먼트에 대한
                        ... 자세한 정보 표시
  
```

# 네트워크 관리 툴

- netsh

- 많은 기능을 가진 네트워크 관리 명령어
- NIC에 IP 주소 할당
  - 'netsh interface ip set address name="이더넷 5" static 192.168.50.10 255.255.255.0 192.168.50.1'
  - 관리자 모드에서 실행 필요
- UNIX계열 컴퓨터에서는 ifconfig 명령어 사용
- 'ifconfig "이더넷 5" 192.168.50.10 255.255.255.0 192.168.50.1'

# 네트워크 관리 툴

## ● netstat

- 컴퓨터의 실제 네트워크 연결상태를 보여줌

```
C:\Users\nhjung>netstat
활성 연결
```

| 프로토콜 | 로컬 주소                              | 외부 주소                  | 상태          |
|------|------------------------------------|------------------------|-------------|
| TCP  | 127.0.0.1:5354                     | NHJUNG-HOME2:8586      | ESTABLISHED |
| TCP  | 127.0.0.1:5354                     | NHJUNG-HOME2:8591      | ESTABLISHED |
| TCP  | 127.0.0.1:5756                     | NHJUNG-HOME2:65001     | ESTABLISHED |
| TCP  | 127.0.0.1:6436                     | NHJUNG-HOME2:27015     | ESTABLISHED |
| TCP  | 127.0.0.1:8586                     | NHJUNG-HOME2:5354      | ESTABLISHED |
| TCP  | 127.0.0.1:8591                     | NHJUNG-HOME2:5354      | ESTABLISHED |
| TCP  | 127.0.0.1:27015                    | NHJUNG-HOME2:6436      | ESTABLISHED |
| TCP  | 127.0.0.1:65001                    | NHJUNG-HOME2:5756      | ESTABLISHED |
| TCP  | 192.168.0.14:4598                  | 52.163.231.110:https   | ESTABLISHED |
| TCP  | 192.168.0.14:4688                  | 52.111.232.10:https    | ESTABLISHED |
| TCP  | 192.168.0.14:4853                  | nrt20s02-in-f10:https  | CLOSE_WAIT  |
| TCP  | 192.168.0.14:4856                  | nrt20s02-in-f10:https  | CLOSE_WAIT  |
| TCP  | 192.168.0.14:4858                  | nrt20s02-in-f10:https  | CLOSE_WAIT  |
| TCP  | 192.168.0.14:4872                  | nrt20s02-in-f10:https  | CLOSE_WAIT  |
| TCP  | 192.168.0.14:4896                  | 20.198.162.76:https    | ESTABLISHED |
| TCP  | 192.168.0.14:6366                  | nrt13s50-in-f74:https  | ESTABLISHED |
| TCP  | 192.168.0.14:7575                  | a23-40-44-62:https     | CLOSE_WAIT  |
| TCP  | 192.168.0.14:8440                  | tl-in-f188:5228        | ESTABLISHED |
| TCP  | 192.168.0.14:8443                  | nrt13s49-in-f238:https | CLOSE_WAIT  |
| TCP  | 192.168.0.14:8444                  | 52.159.49.199:https    | ESTABLISHED |
| TCP  | 192.168.0.14:8445                  | 52.159.49.199:https    | ESTABLISHED |
| TCP  | 192.168.0.14:8449                  | 52.114.20.18:https     | ESTABLISHED |
| TCP  | 192.168.0.14:8450                  | nrt20s19-in-f13:https  | TIME_WAIT   |
| TCP  | 192.168.0.14:8451                  | nrt20s20-in-f14:https  | TIME_WAIT   |
| TCP  | 192.168.0.14:8452                  | ec2-3-35-46-245:49478  | TIME_WAIT   |
| TCP  | 192.168.0.14:8453                  | ec2-3-35-46-245:49478  | TIME_WAIT   |
| TCP  | 192.168.0.14:8454                  | 52.139.153.205:https   | TIME_WAIT   |
| TCP  | 192.168.0.14:8455                  | 52.109.52.78:https     | TIME_WAIT   |
| TCP  | 192.168.0.14:8456                  | 20.44.232.74:https     | ESTABLISHED |
| TCP  | 192.168.0.14:8457                  | 52.109.52.19:https     | ESTABLISHED |
| TCP  | 192.168.0.14:11165                 | 20.198.162.76:https    | ESTABLISHED |
| TCP  | [fe80::5a1:15d8:a9b3:2730x15]:8460 | EPSON8AAEB4:http       | ESTABLISHED |

# 네트워크 관리 툴

- ping
  - 연결 테스트
  - 상대 컴퓨터와 연결이 가능한지 테스트
  - `ping` '주소'
- telnet
  - 서버 연결 테스트
  - 상대 컴퓨터에 있는 서버프로그램과 연결이 되는가를 테스트
  - `telnet` '주소' '포트번호'

# 네트워크 관리 툴

- route

- NIC가 여러 개 있을 경우 IP Packet의 전송 경로 설정
- 이 명령어를 통해 컴퓨터를 라우터로 사용할 수 있음.

```
> route ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3 IF 2
      destination^      ^mask      ^gateway      metric^      ^
                        Interface^
IF를 지정하지 않으면 지정된 게이트웨이에 가장 적절한 인터페이스를 찾으려고
합니다.
> route ADD 3ffe::/32 3ffe::1
> route CHANGE 157.0.0.0 MASK 255.0.0.0 157.55.80.5 METRIC 2 IF 2
CHANGE는 게이트웨이 및/또는 메트릭을 수정하는 데만 사용됩니다.
> route DELETE 157.0.0.0
> route DELETE 3ffe::/32
```

# 네트워크 관리 툴

## ● route

- NIC가 여러 개 있을 경우 IP Packet의 전송 경로 설정
- 이 명령어를 통해 컴퓨터를 라우터로 사용할 수 있음.

```

> route ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3 IF 2
      destination^      ^mask      ^gateway      metric^      ^
                        Interface^
IF를 지정하지 않으면 지정된 게이트웨이에 가장 적절한 인터페이스를 찾으려고
합니다.
> route ADD 3ffe::/32 3ffe::1
> route CHANGE 157.0.0.0 MASK 255.0.0.0 157.55.80.5 METRIC 2 IF 2
CHANGE는 게이트웨이 및/또는 메트릭을 수정하는 데만 사용됩니다.
> route DELETE 157.0.0.0
> route DELETE 3ffe::/32
  
```



# 네트워크 관리 툴

---

- `tracert`
  - 목적지까지의 경로를 추적해 줌

## 과제 2 리뷰

- 노드 A에서 노드 B, C, D로 선택적으로 문자열을 보냄
  - 네트워크에 주소와 가변 길이 개념 추가

# 중간고사

- 4월 23일 금요일 수업시간
  - 대면 => 강의실 이전 예정
  - 시험범위 : 수업 진도 나간 곳 까지
  - 시험 : 강의 내용. 숙제 내용