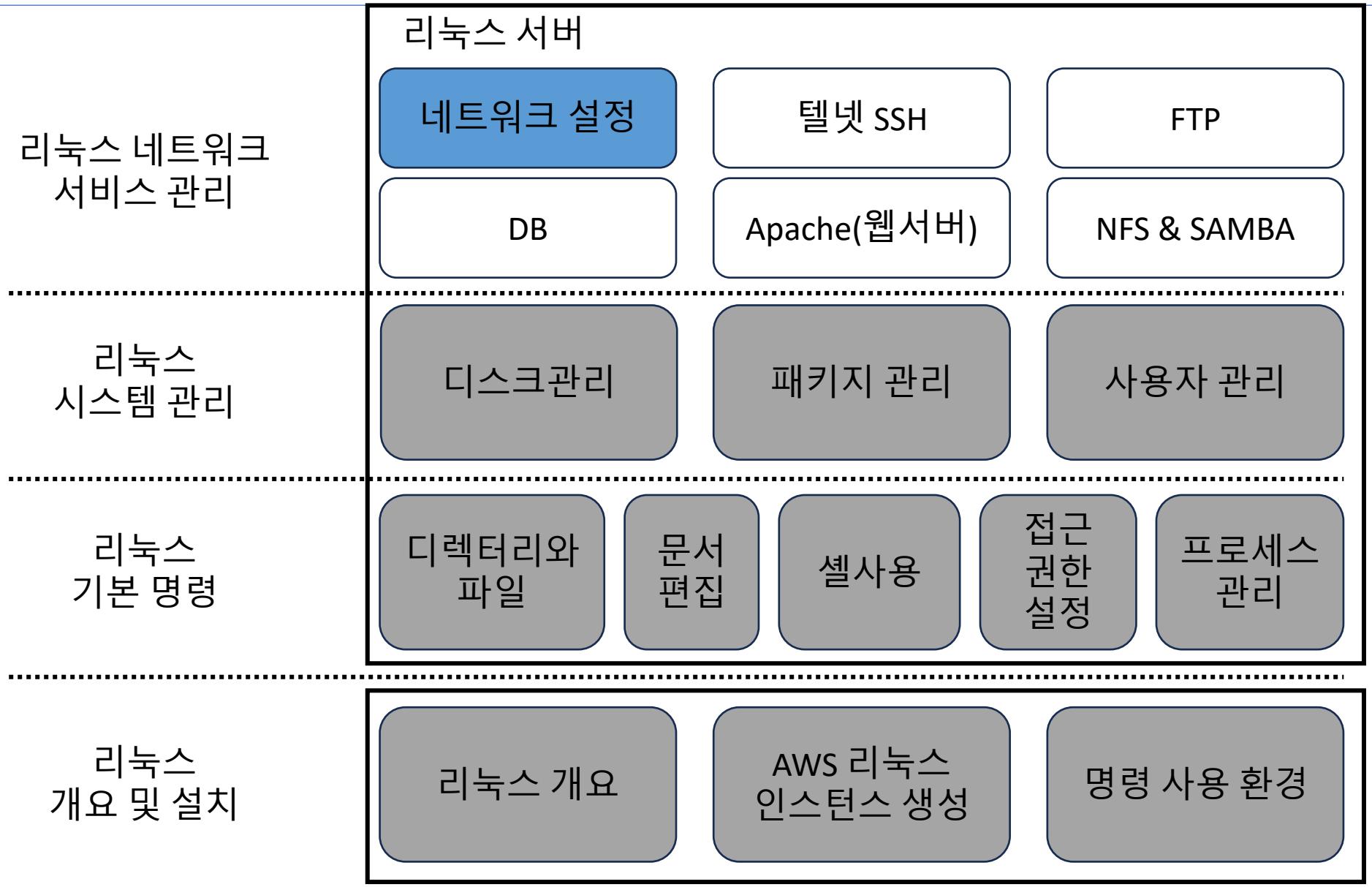
인하공업전문대학
INHA TECHNICAL COLLEGE

서버구축실습

2023년 2학기

서버 구축 실습 STACK



01 네트워크 기초

01. 네트워크 기초

TCP/IP 프로토콜

- 프로토콜: 컴퓨터와 컴퓨터 사이에 데이터를 어떻게 주고받을 것인지를 정의한 통신 규약
- 인터넷은 TCP/IP라는 프로토콜에 따라 통신

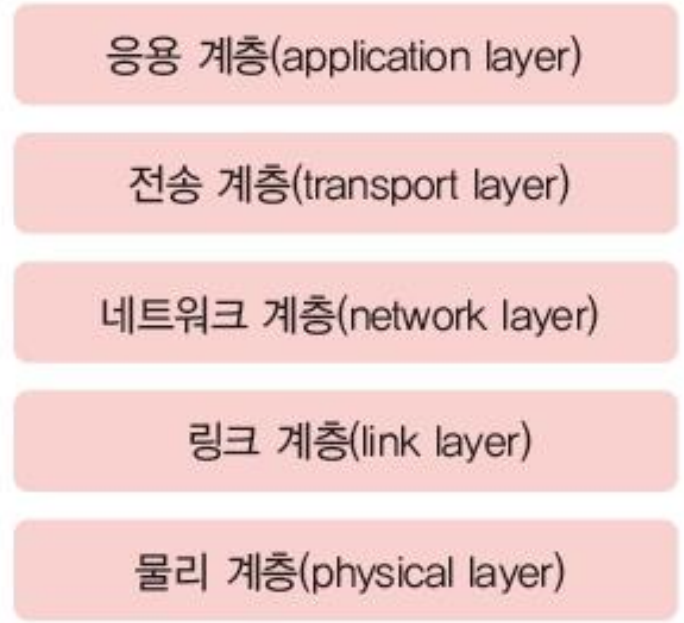


그림 11-1 TCP/IP 프로토콜 모델

표 11-1 TCP/IP 프로토콜 모델의 계층별 역할과 대표 프로토콜

계층	기능	프로토콜	전송 단위
응용 계층	서비스 제공 응용 프로그램	DNS, FTP, SSH, HTTP, 텔넷	메시지
전송 계층	응용 프로그램으로 데이터 전달, 데이터 흐름 제어 및 전송 신뢰성 담당	TCP, UDP	세그먼트
네트워크 계층	주소 관리 및 경로 탐색	IP, ICMP	패킷
링크 계층	네트워크 장치 드라이버	ARP	프레임
물리 계층	케이블 등 전송 매체	구리선, 광케이블, 무선	비트

01. 네트워크 기초

주소

- 컴퓨터의 주소: MAC 주소, IP 주소, 호스트명

MAC 주소

- 하드웨어를 위한 주소
- 이더넷 주소, 하드웨어 주소, 물리 주소라고도 함
- MAC 주소는 네트워크 인터페이스 카드(랜 카드)에 저장된 주소로 기본적으로 네트워크 인터페이스 카드가 만들어질 때 부여
- MAC 주소는 각 하드웨어를 구별하는 역할을 수행
- MAC 주소는 : 이나 - 으로 구분되는 여섯 개의 16진수로 구성되며 총 48bit
  - 앞의 세 자리는 제조사 번호이고 뒤의 세 자리는 일련번호
  - 제조사 번호는 국제 표준 기구 중 하나인 IEEE에서 지정

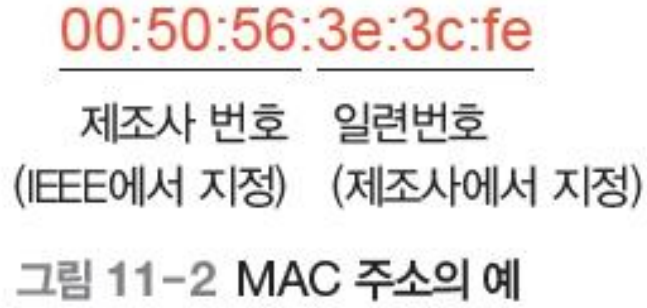


그림 11-2 MAC 주소의 예

01. 네트워크 기초

IP 주소

그림 11-4 IP 주소의 구성

- 인터넷으로 연결된 네트워크에서 각 컴퓨터를 구분하기 위해 사용
- IP 주소는 1바이트 크기의 숫자 네 개로 구성되므로 총 4바이트
- TCP/IP 프로토콜의 3~5계층은 IP 주소를 사용
- IP주소는 네트워크를 구분하는 네트워크 주소 부분과, 해당 네트워크 안에서 특정 컴퓨터를 식별하는 호스트 주소 부분
- IP 주소는 총 32bit(4B) 중 몇 비트를 네트워크 부분으로 사용하고 나머지 몇 비트를 호스트 부분으로 사용하는지에 따라 A 클래스, B 클래스, C 클래스로 구분
- C 클래스는 앞의 3바이트가 네트워크 부분이고 뒤의 1바이트만 호스트 부분
  - 호스트 부분으로 사용할 수 있는 숫자는 0~255인데, 0은 네트워크 주소를 나타내는 데 사용하고 255는 브로드캐스트 주소로 사용하므로 1~254를 호스트 주소로 할당
- IPv6(IP 버전 6)는 fe80::250:56ff:fe3e:3cfe와 같이 16진수로 표기하며 128bit 크기

01. 네트워크 기초

■ 넷마스크와 브로드캐스트 주소

- 넷마스크: IP 주소에서 네트워크 부분을 알려주는 역할

그림 11-5 넷 마스크 계산의 예

- 브로드캐스트 주소: 같은 네트워크에 있는 모든 컴퓨터에 메시지를 보낼 때 사용
  - 호스트 부분을 모두 1로 설정
  - 예: IP 주소에서 네트워크 부분이 192.168.100.0이면 브로드캐스트 주소는 192.168.100.255

01. 네트워크 기초

■ 호스트 이름

- 컴퓨터에 붙이는 이름
- 호스트 이름도 IP 주소처럼 네트워크 부분과 호스트 이름 두 부분으로 구성
- 예: www.naver.com, 호스트 부분(www), 네트워크 부분(naver.com)

■ 포트 번호

- 서버에서 제공하는 각 서비스를 구분하는 번호(예: 웹 - 80)
- /etc/services 파일에 서비스 포트번호 저장

```
tcpmux          1/tcp                           # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
(생략)
ftp             21/tcp
fsp             21/udp          fspd
ssh             22/tcp                          # SSH Remote Login Protocol
(생략)
```

02 네트워크 설정

02. 네트워크 설정

네트워크 설정에 필요한 정보

- IP 주소
- 넷마스크와 브로드캐스트 주소
- 게이트웨이(라우터) 주소
- DNS 주소

02. 네트워크 설정

■ 네트워크 관리자

- 네트워크의 제어와 설정을 관리하는 데몬

■ 네트워크 관리 도구

표 11-2 네트워크 관리 도구

도구	기능
네트워크 관리자	기본 네트워킹 데몬
nmcli 명령	네트워크 관리자를 사용하는 명령 기반 도구
[설정]-[네트워크]	그놈에서 제공하는 GUI 기반 도구
nm-connection-editor	네트워크 관리자를 사용하는 GUI 기반 도구로, [제어판]-[네트워크]에서 설정할 수 없는 부분도 설정할 수 있다.
ip 명령	네트워크를 설정하는 명령을 제공한다.

02. 네트워크 설정

- 네트워크 관리자 설치하기

```
sudo apt install network-manager
```

- 네트워크 관리자 실행하기: 시스템이 부팅할 때 자동으로 동작

```
user1@myubuntu:~$ systemctl status NetworkManager
● NetworkManager.service - Network Manager
     Loaded: loaded (/lib/systemd/system/NetworkManager.service; enabled; vendor
preset: enabl>
     Active: active (running) since Wed 2022-01-05 14:43:42 KST; 4h 56min ago
       Docs: man:NetworkManager(8)
```

  - 네트워크 관리자의 상태가 inactive라면 다음 명령으로 동작

```
sudo systemctl start NetworkManager
```

```
sudo systemctl enable NetworkManager
```

02. 네트워크 설정

■ nmcli 명령으로 네트워크 설정

nmcli

- **기능** 명령 기반으로 네트워크 관리자를 설정한다.
- **형식** nmcli [옵션] 명령 {서브 명령}
- **옵션** -t: 실행 결과를 간단하게 출력한다.
  -p: 사용자가 읽기 좋게 출력한다.
  -v: nmcli의 버전을 출력한다.
  -h: 도움말을 출력한다.

- **명령 {서브 명령}** general {status ¦ hostname}: 네트워크 관리자의 전체적인 상태를 출력하고, 호스트명을 읽거나 변경할 수 있다.
  networking {on ¦ off ¦ connectivity}: 네트워크를 시작 · 종료하고 연결 상태를 출력한다.
  connection {show ¦ up ¦ down ¦ modify ¦ add ¦ delete ¦ reload ¦ load }: 네트워크를 설정한다.
  device {status ¦ show}: 네트워크 장치의 상태를 출력한다.
- **사용 예**

```
nmcli general
nmcli networking on
nmcli con add type ethernet con-name test-net ifname ens33 ip4
192.168.1.10/24 gw4 192.168.1.254
```

02. 네트워크 설정

- 네트워크의 전체 상태 살펴보기: general(gen) 명령

```
user1@myubuntu:~$ nmcli general status
STATE      CONNECTIVITY  WIFI-HW  WIFI  WWAN-HW  WWAN
연결됨   전체            사용      사용  사용      사용
user1@myubuntu:~$ nmcli gen
STATE      CONNECTIVITY  WIFI-HW  WIFI  WWAN-HW  WWAN
연결됨   전체            사용      사용  사용      사용
```

02. 네트워크 설정

- 네트워크를 활성화하거나 비활성화하기: networking(net) 명령

```
user1@myubuntu:~$ nmcli net con
full
user1@myubuntu:~$ nmcli net off
user1@myubuntu:~$ nmcli net con
none
user1@myubuntu:~$ nmcli net on
user1@myubuntu:~$ nmcli net con
full
```

- none(없음): 호스트가 아직 네트워크에 연결되어 있지 않다.
- limited(제한적): 호스트가 네트워크에 연결되어 있지만 인터넷과 연결되지는 않았다.
- full(전체): 호스트가 네트워크에 연결되어 있고 인터넷도 사용할 수 있다.
- unknown(알 수 없음): 네트워크 연결 상태를 알 수 없다.

네트워크 연결 활성화/비활성화

02. 네트워크 설정

- 네트워크 설정하기: connetion(con) 명령

표 11-3 connection의 서브 명령

서브 명령	기능
show	메모리와 디스크에 저장된 네트워크 연결 프로파일을 출력한다. 서브 명령을 지정하지 않을 경우 기본적으로 show를 실행한다.
up	네트워크 연결을 시작한다.
down	네트워크 연결을 중지한다.
modify	연결 프로파일에서 속성을 추가 · 수정 · 삭제한다.
add	새로운 연결을 생성한다.
delete	연결의 설정을 삭제한다.
reload	연결과 관련된 파일을 디스크에서 다시 읽어온다.
load	디스크에서 하나 이상의 연결 파일을 읽어온다.

02. 네트워크 설정

- 네트워크 연결 프로파일 출력하기: show
  - 연결 프로파일의 이름과 UUID, 네트워크 유형, 연결된 장치명을 출력

```
user1@myubuntu:~$ nmcli con show
NAME          UUID                                  TYPE      DEVICE
유선 연결  1  aa04b119-ab6c-3456-ad92-19d7aa5a5985  ethernet  ens33
```

  - ens33은 실제로 외부와 통신할 때 사용하는 네트워크 인터페이스의 명칭

02. 네트워크 설정

- 네트워크 연결 중지/시작 하기: down, up

```
ser1@myubuntu:~$ nmcli con down '유선 연결 1'
연결 '유선 연결 1'이(가) 성공적으로 비활성화되었습니다(D-Bus 활성 경로: /org/freedesktop/
NetworkManager/ActiveConnection/3).
user1@myubuntu:~$ nmcli con show
NAME         UUID                                  TYPE      DEVICE
유선 연결  1  aa04b119-ab6c-3456-ad92-19d7aa5a5985  ethernet  --
```

```
ser1@myubuntu:~$ nmcli con up '유선 연결 1'
연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로: /org/freedesktop/NetworkManager/
ActiveConnection/4)
user1@myubuntu:~$ nmcli con show
NAME         UUID                                  TYPE      DEVICE
유선 연결 1  aa04b119-ab6c-3456-ad92-19d7aa5a5985  ethernet  ens33
```

02. 네트워크 설정

- 네트워크 연결 추가하기: add

연결 추가 형식

```
nmcli connection add type ethernet con-name connection-name ifname interface-name
ip4 address gw4 address
```

연결 추가 예

```
user1@myubuntu:~$ sudo nmcli con add type ethernet con-name test-net ifname ens33 ip4
192.168.1.10/24 gw4 192.168.1.1
연결 'test-net' (56fab3b2-abaf-41a5-9ac8-accc1b9546d7)이 성공적으로 추가되었습니다.
```

연결 추가 확인

```
user1@myubuntu:~$ nmcli con show
NAME          UUID                                  TYPE      DEVICE
유선 연결 1   aa04b119-ab6c-3456-ad92-19d7aa5a5985  ethernet  ens33
test-net      56fab3b2-abaf-41a5-9ac8-accc1b9546d7  ethernet  --
```

연결 활성화

```
user1@myubuntu:~$ sudo nmcli con up test-net ifname ens33
연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로: /org/freedesktop/NetworkManager/
ActiveConnection/5)
```

02. 네트워크 설정

- 네트워크 연결 수정하기: modify(mod)

```
nmcli connection modify connection-name setting.property value
```

표 11-4 설정과 속성의 예

설정(setting)	속성(property)	값의 유형	기능
connection	autoconnection	boolean (TRUE/FALSE)	자원이 사용 가능해지면 네트워크 관리자가 자동으로 연결할지를 지정한다.
	id	문자열	사용자가 읽을 수 있는 연결의 이름
	interface-name	문자열	네트워크 장치의 이름
	type	문자열	연결의 유형
ipv4	addresses	주소	IP 주소
	dns	주소	DNS 서버의 IP 주소
	gateway	주소	게이트웨이 주소
	method	문자열	IP 구성 방법으로 manual은 고정 IP 사용, auto는 동적 IP 사용을 의미
	routes	주소	네트워크의 경로를 설정한다. (예) ipv4.routes "192.168.1.0/24 192.168.1.1").

02. 네트워크 설정

- 예: 기존 test-net 연결의 IPv4 주소를 변경

```
nmcli con mod test-net ipv4.addresses 192.168.122.131
```

- 예: 다른 IP 주소를 추가하려면 +기호를 사용하고, 주소를 제거하려면 -기호를 사용

```
nmcli con mod test-net +ipv4.addresses 192.168.122.132
```

- 게이트웨이 수정

```
nmcli con mod test-net ipv4.gateway 192.168.122.254
```

- 특정 네트워크로 가는 경로를 지정

```
nmcli con mod test-net +ipv4.routes "192.168.2.0/24 192.168.122.1"
```

- 연결 프로파일의 내용을 수정하면 up 명령을 사용해 다시 적용

```
nmcli con up test-net ifname ens33
```

02. 네트워크 설정

- 네트워크 연결 삭제하기: delete(del)

```
nmcli connection delete connection-name
```

```
user1@myubuntu:~$ nmcli con delete test-net
'test-net'(56fab3b2-abaf-41a5-9ac8-accc1b9546d7) 연결이 성공적으로 삭제되었습니다.
user1@myubuntu:~$ nmcli con show
NAME         UUID                                  TYPE      DEVICE
유선 연결 1   aa04b119-ab6c-3456-ad92-19d7aa5a5985   ethernet  ens33
```

- 네트워크 연결 프로파일 읽어오기: reload, load

```
nmcli connection reload
nmcli connection load connection-name
```

02. 네트워크 설정

- 네트워크 장치 상태보기: device(dev) 명령
  - 네트워크 장치의 상태 보기: status

```
user1@myubuntu:~$ nmcli dev status
DEVICE  TYPE      STATE           CONNECTION
ens33   ethernet  연결됨          유선 연결 1
lo      loopback  관리되지 않음   --
```

  - 네트워크 장치의 상세한 정보 보기: show

```
ser1@myubuntu:~$ nmcli dev show
GENERAL.DEVICE:                         ens33
GENERAL.TYPE:                           ethernet
GENERAL.HWADDR:                         00:0C:29:3B:B8:96
GENERAL.MTU:                            1500
GENERAL.STATE:                          100 (연결됨)
GENERAL.CONNECTION:                     유선 연결 1
GENERAL.CON-PATH:                       /org/freedesktop/NetworkManager/
ActiveConnection/6
```

  - 특정 장치만 보려면 장치명 지정: nmcli dev show ens33

02. 네트워크 설정

ip 명령으로 네트워크 설정

ip

- **기능** IP 주소, 게이트웨이, 네트워크 장치의 상태 등을 출력하고 관리한다.
- **형식** ip [옵션] 객체 [서브 명령]
- **옵션** -V: 버전을 출력한다.
  -s: 자세한 정보를 출력한다.
- **객체[서브명령]** address [add|del|show|help]: 장치의 IP 주소를 관리한다(ip-address).
  route [add|del|help]: 라우팅 테이블을 관리한다(ip-route).
  link [set]: 네트워크 인터페이스를 활성화·비활성화한다.
- **사용 예**

```
ip addr show
ip addr add 192.168.1.20/24 dev ens33
ip route show
ip route add 192.168.2.0/24 via 192.168.1.1 dev ens33
```

02. 네트워크 설정

- 네트워크 장치의 주소 관리하기: address(addr) 명령
  - 네트워크 장치의 정보 보기: show

```
user1@myubuntu:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 00:0c:29:3b:b8:96 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.147.130/24 brd 192.168.147.255 scope global dynamic noprefixroute
ens33
       valid_lft 1596sec preferred_lft 1596sec
    inet6 fe80::3886:14f8:fc88:94aa/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

show 명령에 특정 장치를 지정하면 해당 장치의 정보만 출력:

ip addr show ens33

02. 네트워크 설정

- IP 주소 설정하기: add

```
user1@myubuntu:~$ sudo ip addr add 192.168.1.20/24 dev ens33
```

```
user1@myubuntu:~$ ip addr show ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 00:0c:29:3b:b8:96 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.147.130/24 brd 192.168.147.255 scope global dynamic noprefixroute
ens33
       valid_lft 1500sec preferred_lft 1500sec
    inet 192.168.1.20/24 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::3886:14f8:fc88:94aa/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

ens33에 주소 2개 설정

02. 네트워크 설정

- IP 주소 삭제하기: del

```
user1@myubuntu:~$ sudo ip addr del 192.168.1.20/24 dev ens33
user1@myubuntu:~$ ip addr show ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 00:0c:29:3b:b8:96 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.147.130/24 brd 192.168.147.255 scope global dynamic noprefixroute
ens33
       valid_lft 1454sec preferred_lft 1454sec
    inet6 fe80::3886:14f8:fc88:94aa/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

02. 네트워크 설정

■ 라우팅 테이블과 게이트웨이 주소 관리하기: route 명령

- 게이트웨이: 네트워크를 다른 네트워크와 연결할 때 연결점이 되는 장치
- 게이트웨이도 하나의 컴퓨터로 보통 라우터라고 함
- 게이트웨이 주소가 설정되어 있지 않으면 같은 네트워크가 아닌 컴퓨터와는 접속할 수 없음

- 라우팅 테이블 보기: show

```
user1@myubuntu:~$ ip route show
default via 192.168.147.2 dev ens33 proto dhcp metric 100
169.254.0.0/16 dev ens33 scope link metric 1000
192.168.147.0/24 dev ens33 proto kernel scope link src 192.168.147.130 metric 100
```

02. 네트워크 설정

- 기본 게이트웨이 주소 설정하기: add

```
sudo ip route add default via 192.168.1.1 dev ens33
```

- 라우팅 경로 설정하기: add

```
user1@myubuntu:~$ sudo ip route add 192.168.2.0/24 via 192.168.147.2 dev ens33
user1@myubuntu:~$ ip route show
default via 192.168.147.2 dev ens33 proto dhcp metric 100
169.254.0.0/16 dev ens33 scope link metric 1000
192.168.2.0/24 via 192.168.147.2 dev ens33
192.168.147.0/24 dev ens33 proto kernel scope link src 192.168.147.130 metric 100
```

- 라우팅 경로 삭제하기: del

```
user1@myubuntu:~$ sudo ip route del 192.168.2.0/24
```

02. 네트워크 설정

- 네트워크 인터페이스를 활성화하거나 비활성화하기: link 명령
  - 네트워크 인터페이스 비활성화: down

```
user1@myubuntu:~$ sudo ip link set ens33 down
user1@myubuntu:~$ ip addr show ens33
2: ens33: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel state DOWN group default
qlen 1000
    link/ether 00:0c:29:3b:b8:96 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
```

  - 네트워크 인터페이스 활성화: up

```
user1@myubuntu:~$ sudo ip link set ens33 up
user1@myubuntu:~$ ip addr show ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
```

02. 네트워크 설정

ifconfig 명령으로 네트워크 설정

ifconfig

- 기능 네트워크 인터페이스의 IP 주소를 설정한다.
- 형식 ifconfig [인터페이스명] [옵션] [값]
- 옵션 -a: 시스템의 전체 인터페이스에 대한 정보를 출력한다.
  up/down: 인터페이스를 활성화 · 비활성화한다.
  netmask 주소: 넷마스크 주소를 설정한다.
  broadcast 주소: 브로드캐스트 주소를 설정한다.
- 사용 예 ifconfig ens33
  ifconfig ens33 192.168.1.2 netmask 255.255.255.0 broadcast 192.168.1.255

- 우분투에서 ifconfig 명령을 사용하려면 net-tools 패키지를 설치

```
user1@myubuntu:~$ sudo apt install net-tools
```

02. 네트워크 설정

- 현재 설치된 네트워크 인터페이스 설정 보기

```
user1@myubuntu:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.147.130  netmask 255.255.255.0  broadcast 192.168.147.255
        inet6 fe80::3886:14f8:fc88:94aa  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:3b:b8:96  txqueuelen 1000  (Ethernet)
        RX packets 78522  bytes 108991067 (108.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 38009  bytes 2672779 (2.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
```

lo : 로컬 루프백(시스템 내부 통신용)

02. 네트워크 설정

- 특정 네트워크 인터페이스 설정보기

```
user1@myubuntu:~$ ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.147.130  netmask 255.255.255.0  broadcast 192.168.147.255
        inet6 fe80::3886:14f8:fc88:94aa  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:3b:b8:96  txqueuelen 1000  (Ethernet)
        RX packets 78592  bytes 108997859 (108.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 38082  bytes 2680713 (2.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions
```

02. 네트워크 설정

- 네트워크 인터페이스 사용 해제하기: down 옵션

```
user1@myubuntu:~$ sudo ifconfig ens33 down
user1@myubuntu:~$ ifconfig ens33
ens33: flags=4098<BROADCAST,MULTICAST>  mtu 1500
        inet 192.168.147.130  netmask 255.255.255.0  broadcast 192.168.147.255
        ether 00:0c:29:3b:b8:96  txqueuelen 1000  (Ethernet)
        RX packets 78597  bytes 108998247 (108.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 38086  bytes 2681640 (2.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

02. 네트워크 설정

- 네트워크 인터페이스 활성화하기: up 옵션

```
user1@myubuntu:~$ sudo ifconfig ens33 up
user1@myubuntu:~$ ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.147.130  netmask 255.255.255.0  broadcast 192.168.147.255
        inet6 fe80::3886:14f8:fc88:94aa  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:3b:b8:96  txqueuelen 1000  (Ethernet)
        RX packets 78623  bytes 109002421 (109.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 38128  bytes 2687039 (2.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

02. 네트워크 설정

- 네트워크 인터페이스 설정하기

```
ifconfig 인터페이스명 IP 주소 netmask 넷마스크 주소 broadcast 브로드캐스트 주소
```

```
user1@myubuntu:~$ sudo ifconfig ens33 192.168.147.129 netmask 255.255.255.0 broadcast
192.168.147.255
user1@myubuntu:~$ ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.147.129  netmask 255.255.255.0  broadcast 192.168.147.255
        inet6 fe80::3886:14f8:fc88:94aa  prefixlen 64  scopeid 0x20<link>
```

02. 네트워크 설정

■ 게이트웨이 설정하기

route

- **기능** 라우팅 테이블을 편집하고 출력한다.
- **형식** route [명령]
- **명령** add: 라우팅 경로나 기본 게이트웨이를 추가한다.
  del: 라우팅 경로나 기본 게이트웨이를 삭제한다.
- **사용 예** route
  route add default gw 192.168.1.1 dev ens33

02. 네트워크 설정

표 11-5 route 명령을 사용한 라우팅 테이블 편집

기능	명령 형식과 사용 예
라우팅 경로 추가(네트워크)	route add –net 네트워크 주소 netmask 넷마스크 dev 인터페이스명 route add –net 192.168.1.0 netmask 255.255.255.0 dev ens33
라우팅 경로 추가(호스트)	route add –host 호스트 주소 dev 인터페이스명 route add –host 192.168.1.5 dev ens33
라우팅 경로 제거(네트워크)	route del –net 네트워크 주소 netmask 넷마스크 [dev 인터페이스명] route del –net 192.168.1.0 netmask 255.255.255.0
라우팅 경로 제거(호스트)	route del –host 호스트 주소 route del –host 192.168.1.5
기본 게이트웨이 추가	route add default gw 게이트웨이 주소 dev 인터페이스명 route add default gw 192.168.1.1 dev ens33
기본 게이트웨이 제거	route del default gw 게이트웨이 주소 route del default gw 192.168.1.1
루프백(lo) 추가	route add –net 127.0.0.0 netmask 255.0.0.0 dev lo

02. 네트워크 설정

- 라우팅 테이블 보기: route

```
user1@myubuntu:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 ens33
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 ens33
192.168.147.0   0.0.0.0         255.255.255.0   U     100    0        0 ens33
```

02. 네트워크 설정

표 11-6 라우팅 테이블의 출력 항목

항목	기능
Destination	라우팅 대상 네트워크나 호스트의 주소
Gateway	게이트웨이 주소 또는 설정되어 있지 않으면 * 출력
Genmask	대상 네트워크의 넷마스크 255.255.255.255 : 대상이 호스트인 경우 0.0.0.0 : 기본(default) 경로
Flags	U: 경로 활성화(UP) H: 대상이 호스트 G: 게이트웨이로 사용 R: 동적 라우팅을 위한 경로 재생성 D: 데몬 또는 리다이렉트에 의해 동적으로 재설치 M: 라우팅 데몬 또는 리다이렉트에 의해 경로 수정 A: addrconf에 의해 설치 C: 캐시 항목 !: 경로 거부

항목	기능
Metrics	대상까지의 거리로 최근 커널에서는 사용되지 않지만 라우팅 데몬에서 사용할 수도 있다.
Ref	해당 경로에 대한 참조 수이지만 리눅스 커널에서는 사용하지 않는다.
Use	경로를 탐색한 수
Iface	패킷이 전달되는 인터페이스 이름

02. 네트워크 설정

- 기본 게이트웨이 설정하기: add

```
user1@myubuntu:~$ sudo route add default gw 192.168.147.2 dev ens33
user1@myubuntu:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    0      0        0 ens33
default         _gateway        0.0.0.0         UG    100    0        0 ens33
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 ens33
192.168.147.0   0.0.0.0         255.255.255.0   U     100    0        0 ens33
```

02. 네트워크 설정

- 기본 게이트웨이 삭제하기: del

```
user1@myubuntu:~$ sudo route del default gw 192.168.147.2
user1@myubuntu:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 ens33
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 ens33
192.168.147.0   0.0.0.0         255.255.255.0   U     100    0        0 ens33
```

02. 네트워크 설정

DNS 설정

- DNS: 호스트명을 IP 주소로 바꾸는 역할을 수행

DNS 서버 지정하기

- 우분투는 17.04 버전부터 DNS를 관리하는 systemd-resolved 서비스를 도입
- DNS 서버의 주소를 /etc/resolv.conf 파일에 저장

```
user1@myubuntu:~$ cat /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
(생략)
nameserver 127.0.0.53
options edns0 trust-ad
search localdomain
```

02. 네트워크 설정

■ DNS 서버의 정보는 systemd-resolve 명령으로 확인

systemd-resolve

- **기능** DNS 서버에 질의하고 응답을 받는다.
- **형식** systemd-resolve [옵션] [호스트명|IP 주소]
- **옵션** --status: DNS 서버 정보를 출력한다.
- **사용 예** systemd-resolve www.daum.net
  systemd-resolve --status

02. 네트워크 설정

- systemd-resolve --status 명령으로 DNS 서버 정보를 확인

```
user1@myubuntu:~$ systemd-resolve --status
Global
       Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
resolv.conf mode: stub

Link 2 (ens33)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 192.168.147.2
       DNS Servers: 192.168.147.2
        DNS Domain: localdomain
```

02. 네트워크 설정

nmcli 명령으로 DNS 설정하기

```
nmcli con mod connection-name ipv4.dns DNS주소
```

- 예:

```
user1@myubuntu:~$ sudo nmcli con mod '유선 연결 1' ipv4.dns "8.8.8.8 8.8.4.4"
user1@myubuntu:~$ sudo nmcli con up '유선 연결 1'
연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로: /org/freedesktop/NetworkManager/
ActiveConnection/2)
```

- system-resolve --status로 확인

```
user1@myubuntu:~$ systemd-resolve --status
(생략)
Link 2 (ens33)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 192.168.147.2
         DNS Servers: 192.168.147.2 8.8.8.8 8.8.4.4
         DNS Domain: localdomain
```

02. 네트워크 설정

DNS 서버에 질의하기

nslookup

- 기능 DNS 서버와 대화식으로 질의하고 응답을 받는다.
- 형식 nslookup [도메인명]
- 사용 예 nslookup
  nslookup www.daum.net

```
user1@myubuntu:~$ nslookup
> www.hanbit.co.kr
Server:         127.0.0.53
Address:        127.0.0.53#53
Non-authoritative answer:
Name:   www.hanbit.co.kr
Address: 218.38.58.195
> exit
```

- systemd-resolve 명령으로도 DNS 서버에 질의

```
user1@myubuntu:~$ systemd-resolve www.hanbit.co.kr
www.hanbit.co.kr: 218.38.58.195                -- link: ens33

-- Information acquired via protocol DNS in 7.2ms.
-- Data is authenticated: no; Data was acquired via local or encrypted transport: no
-- Data from: network
```

609p. 따라해보기: 네트워크 설정하기

① 현재 네트워크의 연결 상태를 확인: nmcli gen status
② 연결 프로파일을 확인: nmcli con show
③ ip 명령으로 IP 주소를 확인: ip addr show ens33
④ ip 명령으로 라우팅 테이블을 확인: ip route show
⑤ nslookup 명령으로 www.daum.net의 IP 주소를 확인

03 호스트 이름 설정

03. 호스트 이름 설정

■ 호스트 이름 출력하기: uname -n 명령

uname

- **기능** 시스템 정보를 출력한다.
- **형식** uname [옵션]
- **옵션** -m: 하드웨어 종류를 출력한다.
  -n: 호스트 이름을 출력한다.
  -r: 운영체제의 릴리즈 정보를 출력한다.
  -s: 운영체제의 이름을 출력한다.
  -v: 운영체제의 버전을 출력한다.
  -a: 위의 모든 정보를 출력한다.
- **사용 예** uname -n
  uname -a

```
user1@myubuntu:~$ uname -n
myubuntu
```

```
user1@myubuntu:~$ uname -a
Linux myubuntu 5.13.0-22-generic #22-Ubuntu SMP Fri Nov 5 13:21:36 UTC 2021 x86_64
x86_64 x86_64 GNU/Linux
```

03. 호스트 이름 설정

■ 호스트 이름 출력 및 설정하기: hostname 명령

hostname

- 기능 호스트 이름을 출력하거나 설정한다.
- 형식 hostname [호스트 이름]
- 사용 예 hostname
  hostname mail.han.server

```
user1@myubuntu:~$ hostname
myubuntu
```

```
user1@myubuntu:~$ nmcli gen host
myubuntu
```

```
user1@myubuntu:~$ sudo hostname mail.server
user1@myubuntu:~$ hostname
mail.server
```

호스트 이름 설정

03. 호스트 이름 설정

■ 호스트 이름 검색 및 설정하기: hostnamectl 명령

hostnamectl

- **기능** 호스트 이름을 검색하거나 설정한다.
- **형식** hostnamectl [옵션] [명령]
- **옵션** -h: 도움말을 출력한다.
  --version: 버전을 출력한다.
- **명령** status: 현재 호스트 이름과 관련 정보를 출력한다.
  set-hostname 호스트명: 호스트명을 호스트 이름으로 설정한다.
- **사용 예** hostnamectl
  hostnamectl status
  hostnamectl set-hostname mail.han.server

```
user1@myubuntu:~$ sudo hostnamectl set-hostname myubuntu
user1@myubuntu:~$ hostnamectl
 Static hostname: myubuntu
       Icon name: computer-vm
         Chassis: vm
      Machine ID: 33edce09bc144b6baa199b3bca4a4a71
         Boot ID: b7872061820943c5aa283b808fb45e77
  Virtualization: vmware
Operating System: Ubuntu 21.10
          Kernel: Linux 5.13.0-22-generic
    Architecture: x86-64
 Hardware Vendor: VMware, Inc.
  Hardware Model: VMware Virtual Platform
```

03. 호스트 이름 설정

■ 호스트 이름 파일에 저장하기

- 리눅스에서 호스트 이름을 저장하는 파일은 /etc/hostname

```
user1@myubuntu:~$ cat /etc/hostname
myubuntu
```

615p. 따라해보기: 호스트 이름 설정하기

① **hostname 명령으로 호스트 이름을 yubuntu.server로 설정**

② **hostnamectl 명령으로 호스트 이름과 관련 정보를 확인**

③ **hostnamectl 명령으로 호스트 이름을 myubuntu로 설정**

- sudo hostnamectl set-hostname myubuntu

04 네트워크 상태 확인

04. 네트워크 상태 확인

■ 통신 확인

ping

- **기능** 네트워크 장비에 신호(ECHO_REQUEST)를 보낸다.
- **형식** ping [옵션] [목적지 주소]
- **옵션** -a: 통신이 되면 소리를 낸다.
  -q: 테스트 결과를 지속적으로 보여주지 않고 종합 결과만 출력한다.
  -c 개수: 보낼 패킷 수를 지정한다.
- **사용 예** ping 192.168.1.1
  ping -a www.naver.com

04. 네트워크 상태 확인

- 옵션 없이 사용하기: 패킷은 56B의 크기로 전송(64B = 56B의 데이터 + 8B의 헤더 정보)
  - ctrl+c로 ping 종료

```
user1@myubuntu:~$ ping 192.168.147.2
PING 192.168.147.2 (192.168.147.2) 56(84) bytes of data.
64 bytes from 192.168.147.2: icmp_seq=1 ttl=128 time=0.257 ms
64 bytes from 192.168.147.2: icmp_seq=2 ttl=128 time=0.339 ms
64 bytes from 192.168.147.2: icmp_seq=3 ttl=128 time=0.273 ms
64 bytes from 192.168.147.2: icmp_seq=4 ttl=128 time=0.312 ms
(생략)
64 bytes from 192.168.147.2: icmp_seq=11 ttl=128 time=0.326 ms
64 bytes from 192.168.147.2: icmp_seq=12 ttl=128 time=1.13 ms
^C
--- 192.168.147.2 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11190ms
rtt min/avg/max/mdev = 0.257/0.534/1.125/0.346 ms
```

04. 네트워크 상태 확인

- -q 옵션 사용하기: 아무 메시지도 출력되지 않다가 ctrl+C로 종료하면 통계 정보만 출력

```
user1@myubuntu:~$ ping -q 192.168.147.2
PING 192.168.147.2 (192.168.147.2) 56(84) bytes of data.
^C
--- 192.168.147.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2047ms
rtt min/avg/max/mdev = 0.302/0.369/0.448/0.060 ms
```

04. 네트워크 상태 확인

- -c 옵션 사용하기: 보낼 패킷 수를 지정

```
user1@myubuntu:~$ ping -c 3 192.168.147.2
PING 192.168.147.2 (192.168.147.2) 56(84) bytes of data.
64 bytes from 192.168.147.2: icmp_seq=1 ttl=128 time=0.255 ms
64 bytes from 192.168.147.2: icmp_seq=2 ttl=128 time=0.930 ms
64 bytes from 192.168.147.2: icmp_seq=3 ttl=128 time=0.352 ms
--- 192.168.147.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2019ms
rtt min/avg/max/mdev = 0.255/0.512/0.930/0.297 ms
```

04. 네트워크 상태 확인

- 도메인 이름 사용하기

```
user1@myubuntu:~$ ping www.hanbit.co.kr
PING www.hanbit.co.kr (218.38.58.195) 56(84) bytes of data.
64 bytes from 218.38.58.195 (218.38.58.195): icmp_seq=1 ttl=128 time=5.74 ms
64 bytes from 218.38.58.195 (218.38.58.195): icmp_seq=2 ttl=128 time=4.90 ms
64 bytes from 218.38.58.195 (218.38.58.195): icmp_seq=3 ttl=128 time=4.82 ms
64 bytes from 218.38.58.195 (218.38.58.195): icmp_seq=4 ttl=128 time=5.53 ms
^C
--- www.hanbit.co.kr ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3007ms
rtt min/avg/max/mdev = 4.824/5.248/5.741/0.395 ms
```

04. 네트워크 상태 확인

■ 네트워크 상태 정보 출력

netstat

- **기능** 네트워크의 상태 정보를 출력한다.
- **형식** netstat [옵션]
- **옵션** -a: 모든 소켓 정보를 출력한다.
  -r: 라우팅 정보를 출력한다.
  -n: 호스트명 대신 IP 주소로 출력한다.
  -i: 모든 네트워크 인터페이스 정보를 출력한다.
  -s: 프로토콜별로 네트워크 통계 정보를 출력한다.
  -p: 해당 소켓과 관련된 프로세스의 이름과 PID를 출력한다.
- **사용 예** netstat -rn
  netstat -s

04. 네트워크 상태 확인

- 라우팅 테이블 확인하기: -r 옵션
  - -n 옵션을 함께 사용하면 이름 대신 IP 주소를 출력

```
user1@myubuntu:~$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 ens33
link-local      0.0.0.0         255.255.0.0     U         0 0          0 ens33
192.168.147.0   0.0.0.0         255.255.255.0   U         0 0          0 ens33
user1@myubuntu:~$ netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         192.168.147.2   0.0.0.0         UG        0 0          0 ens33
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 ens33
192.168.147.0   0.0.0.0         255.255.255.0   U         0 0          0 ens33
```

04. 네트워크 상태 확인

- 현재 열려 있는 포트 확인하기: 서비스 포트가 LISTEN 상태인지 검색

```
user1@myubuntu:~$ netstat -an | grep LISTEN
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:25              0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:631                 :::*                    LISTEN
tcp6       0      0 :::25                   :::*                    LISTEN
unix  2      [ ACC ]     STREAM     LISTENING     52368    @/tmp/.ICE-unix/2587
unix  2      [ ACC ]     STREAM     LISTENING     43759    @/tmp/dbus-ktjggEgF
unix  2      [ ACC ]     STREAM     LISTENING     52573    @/tmp/.X11-unix/X0
```

04. 네트워크 상태 확인

- 현재 열려 있는 포트를 사용 중인 프로세스 확인하기: -p 옵션

```
user1@myubuntu:~$ sudo netstat -p
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/
Program name
tcp        0    448 myubuntu:ssh            192.168.147.1:12373     ESTABLISHED
3707/sshd: user1 [p
udp        0      0 myubuntu:bootpc         192.168.147.254:bootps  ESTABLISHED 930/
NetworkManager
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   PID/Program name     Path
unix  2      [ ]         DGRAM                    50065    2455/systemd         /
run/user/1000/systemd/notify
unix  4      [ ]         DGRAM                    29659    1/init               /
run/systemd/notify
(생략)
```

04. 네트워크 상태 확인

- 인터페이스별 네트워크 통계 정보 확인하기: -i 옵션
  - 정상적으로 주고받은 패킷의 개수: RX-OK, TX-OK
  - 송수신 중에 오류가 발생한 패킷의 개수: RX-ERR, RX-DRP, RX-OVR, TX-ERR, TX-DRP, TX-OVR

```
user1@myubuntu:~$ netstat -i
Kernel Interface table
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
ens33     1500     2819      0      0 0          2216      0      0      0 BMRU
lo       65536      783      0      0 0           783      0      0      0 LRU
```

04. 네트워크 상태 확인

- 프로토콜별 네트워크 통계 정보 확인하기: -s 옵션

```
user1@myubuntu:~$ netstat -s
Ip:
    Forwarding: 2
    3293 total packets received
    6 with invalid addresses
    0 forwarded
    0 incoming packets discarded
    3285 incoming packets delivered
    2748 requests sent out
    20 outgoing packets dropped
    2 dropped because of missing route
Icmp:
    151 ICMP messages received
    21 input ICMP message failed
    ICMP input histogram:
        destination unreachable: 141
        echo replies: 10
    161 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 141
        echo requests: 20
IcmpMsg:
        InType0: 10
        InType3: 141
        OutType3: 141
        OutType8: 20
Tcp:
    60 active connection openings
    1 passive connection openings
    23 failed connection attempts
    0 connection resets received
    1 connections established
    2161 segments received
    1610 segments sent out
    60 segments retransmitted
    0 bad segments received
    11 resets sent
Udp:
    916 packets received
    56 packets to unknown port received
    0 packet receive errors
    946 packets sent
    0 receive buffer errors
    0 send buffer errors
    IgnoredMulti: 3
```

04. 네트워크 상태 확인

■ MAC 주소와 IP 주소 확인

arp

- 기능 ARP 캐시 정보를 관리한다.
- 형식 arp [IP 주소]
- 사용 예 arp
  arp 192.168.1.1

```
user1@myubuntu:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.147.254          ether   00:50:56:f0:16:90   C                     ens33
_gateway                 ether   00:50:56:e6:26:01   C                     ens33
192.168.147.1            ether   00:50:56:c0:00:08   C                     ens33
```

```
user1@myubuntu:~$ arp 192.168.147.2
Address                  HWtype  HWaddress           Flags Mask            Iface
_gateway                 ether   00:50:56:e6:26:01   C                     ens33
```

04. 네트워크 상태 확인

■ 패킷 캡처 명령

tcpdump

- **기능** 네트워크상의 트래픽을 덤프한다.
- **형식** tcpdump [옵션]
- **옵션** -c 패킷 수: 지정한 패킷 수만큼 덤프 받고 종료한다.
  -i 인터페이스명: 특정 인터페이스를 지정한다.
  -n: IP 주소를 호스트명으로 바꾸지 않는다.
  -q: 정보를 간단한 형태로 보여준다.
  -X: 패킷의 내용을 16진수와 ASCII로 출력한다.
  -w 파일명: 덤프한 내용을 지정한 파일에 저장한다.
  -r 파일명: 덤프를 저장한 파일에서 읽어온다.
  host 호스트명 또는 주소: 지정한 호스트가 받거나 보낸 패킷만 덤프한다.
  tcp port 번호: 지정한 포트 번호 패킷만 덤프한다.
  ip: IP 패킷만 덤프한다.

- **사용 예**

```
tcpdump
tcpdump -i eth0
tcpdump -i eth0 -w DUMP.out
tcpdump tcp port 22 and host 192.168.0.7
```

04. 네트워크 상태 확인

- 옵션 없이 사용하기: 현재 시스템에서 주고받는 모든 패킷을 캡처하여 패킷의 헤더 부분 정보를 출력

```
user1@myubuntu:~$ sudo tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144 bytes
22:13:39.721133 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq
2696033996:2696034124, ack 428333463, win 501, length 128
22:13:39.721207 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq 128:192, ack 1,
win 501, length 64
22:13:39.721263 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq 192:272, ack 1,
win 501, length 80
22:13:39.721350 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq 272:352, ack 1,
win 501, length 80
```

04. 네트워크 상태 확인

- tcpdump는 ctrl+ 로 종료하면 캡처한 패킷의 개수를 출력하고 종료

```
(생략)
22:13:40.022063 IP 192.168.147.1.12373 > myubuntu.ssh: Flags [.], ack 13744, win
4100, length 0
^C
86 packets captured
148 packets received by filter
0 packets dropped by kernel
```

04. 네트워크 상태 확인

- 캡처할 패킷 개수 지정하기: -c 옵션

```
user1@myubuntu:~$ sudo tcpdump -c 3
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144 bytes
22:15:01.464129 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq
2696054812:2696054940, ack 428334839, win 501, length 128
22:15:01.464203 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq 128:192, ack 1,
win 501, length 64
22:15:01.464259 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq 192:272, ack 1,
win 501, length 80
3 packets captured
21 packets received by filter
0 packets dropped by kernel
```

04. 네트워크 상태 확인

- 캡처한 패킷 정보를 파일로 저장하기: -w 옵션

```
user1@myubuntu:~$ sudo tcpdump -c 3 -w dump.out
tcpdump: listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144
bytes
3 packets captured
6 packets received by filter
0 packets dropped by kernel
```

```
user1@myubuntu:~$ file dump.out
dump.out: pcap capture file, microsecond ts (little-endian) - version 2.4 (Ethernet,
capture length 262144)
user1@myubuntu:~$ cat dump.out
▦ ▦e.a▦{▦▦PV▦
)TE▦▦9@@$▦▦▦▦▦ea▦`= P▦}▦I▦d?▦$▦m▦▦\▦]▦Vrd▦▦O▦b▦▦▦▦,▦▦▦~)▦7▦▦e ▦
```

바이너리 파일이라
내용 확인 불가능

04. 네트워크 상태 확인

- 캡처한 패킷 파일 읽기: -r 옵션

```
user1@myubuntu:~$ sudo tcpdump -r dump.out
reading from file dump.out, link-type EN10MB (Ethernet), snapshot length 262144
22:24:03.685500 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq
2696157100:2696157244, ack 428370583, win 501, length 144
22:24:03.685755 IP 192.168.147.1.12373 > myubuntu.ssh: Flags [.], ack 144, win 4098,
length 0
22:24:17.934272 IP 192.168.147.1.12373 > myubuntu.ssh: Flags [P.], seq 1:65, ack 144,
win 4098, length 64
```

- 만약 권한 거부 오류 발생시

```
$ sudo apparmor_parser -R /etc/apparmor.d/usr.bin.tcpdump
```

04. 네트워크 상태 확인

- 특정 포트로 송수신되는 패킷 캡처하기: tcp port 옵션

22번(ssh) 포트 캡처

```
user1@myubuntu:~$ sudo tcpdump -c 3 tcp port 22
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144 bytes
22:29:41.033056 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq
2696182524:2696182732, ack 428388951, win 501, length 208
22:29:41.033301 IP 192.168.147.1.12373 > myubuntu.ssh: Flags [.], ack 208, win 4100,
length 0
22:29:41.135522 IP myubuntu.ssh > 192.168.147.1.12373: Flags [P.], seq 208:400, ack 1,
win 501, length 192
3 packets captured
10 packets received by filter
0 packets dropped by kernel
```

04. 네트워크 상태 확인

- 캡처한 내용을 ASCII로 보기: -X 옵션

```
user1@myubuntu:~$ sudo tcpdump -Xqr dump.out
reading from file dump.out, link-type EN10MB (Ethernet), snapshot length 262144
22:24:03.685500 IP myubuntu.ssh > 192.168.147.1.12373: tcp 144
        0x0000:  4510 00b8 3a8f 4000 4006 57cc c0a8 9382  E...:.@.@.W.....
        0x0010:  c0a8 9301 0016 3055 a0b4 17ac 1988 6a97  ......0U......j.
        0x0020:  5018 01f5 a87f 0000 472a c955 eeb5 2296  P.......G*.U..".
        0x0030:  44db bd6b 2ff2 fbf5 6644 284e 04f8 2386  D..k/...fD(N..#.
        0x0040:  b141 7259 57ea 80f4 aea1 f76b 8154 a2c6  .ArYW......k.T..
        0x0050:  0832 416e 960a c809 e8f6 f715 6616 a156  .2An........f..V
        0x0060:  e996 9592 09d6 b180 0e1f 9069 d311 cc84  ...........i....
        0x0070:  d616 8a49 2bb3 140e 1e41 c05e d94e a7a0  ...I+....A.^.N..
        0x0080:  33df 0cab 3266 4c06 fcd4 7ee4 17d8 3d49  3...2fL...~...=I
        0x0090:  cf42 9129 2fdd 30ef e28d 64fd 6fb1 7411  .B.)/.0...d.o.t.
        0x00a0:  a460 5a3b f928 3c71 6aa2 2855 3b5f dd32  .`Z;.(<qj.(U;_.2
        0x00b0:  2a82 43a8 bb1d 5605                      *.C...V.
(생략)
```

629p. 따라해보기: 네트워크 상태 확인하기

① ping 명령으로 www.google.co.kr이 동작하는지 확인

② netstat 명령으로 현재 열려 있는 tcp 포트를 사용하는 프로세스를 확인

- sudo netstat -pt

③ 웹 브라우저(파이어폭스)를 동작시키고 www.daum.net 사이트에 접속

④ tcp 포트 80으로 주고받는 패킷 다섯 개를 캡처하여 httpdump.out 파일에 저장

⑤ httpdump.out 파일에 저장한 패킷의 내용을 ASCII로 확인