

# 네트워크와 방화벽

## #01. 네트워크의 이해

### 1. TCP/IP

- 네트워크 전송 프로토콜로서 인터넷 표준 프로토콜(=규약)임
- Transmission Control Protocol 의 줄임말로써 Internet Protocol 이라고도 함

구분	설명
TCP 프로토콜	전송 데이터를 일정 단위(패킷)으로 나누고 포장하는 것에 관한 규약
IP 프로토콜	직접 데이터를 주고 받는 것에 관한 규약

### 2. IP주소

- 네트워크상에 존재하는 컴퓨터의 고유번호
- IPv4 방식과 IPv6 방식이 존재 (IPv6는 잘 사용하지 않음)
- 숫자 네 개로 IP주소를 표현하는데, 각 숫자는 0번부터 255번까지 범위를 가짐
- 로컬 컴퓨터의 주소를 의미하는 IP는 127.0.0.1로 고정되어 있음

IP 클래스	범위	설명
A	0.0.0.0 ~ 127.255.255.255	255.0.0.0
B	128.0.0.0 ~ 191.255.255.255	255.255.0.0
C	192.0.0.0 ~ 223.255.255.255	255.255.255.0 (일반 컴퓨터용)
D	224.0.0.0 ~ 239.255.255.255	멀티캐스트용
E	240.0.0.0 ~ 255.255.255.255	연구용 등으로 예약

#### 1) 공인 IP

- 공인 기관에서 할당받아 사용함.
- 네트워크상에 공개된 유일한 IP 번호로서, 공인 IP를 알면 누구나 쉽게 접속이 가능함.

#### 2) 사설 IP

- 내부적으로만 사용하기 때문에 내부 IP라고도 함.
- 주로 공유기 등의 장비를 통해 할당받는 아이피
- 외부로부터의 접근이 불가능함
- 사설 IP의 대역은 정해져 있음
  - 10.0.0.1 ~ 10.255.255.254
  - 172.16.0.1 ~ 172.31.255.254
  - 192.168.0.1 ~ 192.168.255.254

### 3. 호스트 이름과 도메인 이름



### 4. 서브넷 마스크

- 하나의 네트워크를 구분하는 단위
- IP주소는 네트워크 주소와 호스트 주소로 나눌 수 있음
- IP주소에서 네트워크 주소와 호스트 주소를 구분하는 값으로 사용됨



### 5. 게이트웨이(Gateway)

- 외부 네트워크로 통신을 하기 위한 통로.
- 내부 네트워크에서 외부 네트워크로 연결되는 컴퓨터 혹은 라우터가 게이트웨이 역할을 함.
- 라우터 : 네트워크 트래픽을 포워딩 해야 하는 최적 경로를 결정하는 장비 (공유기)

### 6. DNS

- IP주소를 호스트와 도메인 이름으로 변환해 주는 서비스를 제공하는 서버
- KT가 운영하는 DNS의 경우 168.126.63.1, 168.126.63.2임

## #02. 네트워크 관련 명령어

### 1) IP 주소 확인하기

`ifconfig` 명령으로 IP주소를 확인할 수 있지만 이 명령을 사용하기 위해서는 `net-tools`를 설치해야 한다.

```
$ sudo apt-get update
$ sudo apt-get install net-tools
```

설치가 완료되면 아래의 명령으로 IP주소를 확인할 수 있다.

```
$ ifconfig
```

```
leekh@my-server:~$ ifconfig
enp0s5: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.211.55.8 netmask 255.255.255.0 broadcast 10.211.55.255
    inet6 fdb2:2c26:f4e4:0:21c:42ff:fe50:80c1 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::21c:42ff:fe50:80c1 prefixlen 64 scopeid 0x20<link>
    ether 00:1c:42:50:80:c1 txqueuelen 1000 (Ethernet)
    RX packets 12971 bytes 4065914 (4.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5250 bytes 961380 (961.3 KB)
    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
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 134 bytes 11207 (11.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 134 bytes 11207 (11.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

leekh@my-server:~$
```

`ifconfig` 장치명 형식으로 특정 장치에 대한 정보만 확인할 수 있다

### 2) 네트워크 접속 가능 여부 확인하기

- 1초마다 목적지 컴퓨터로 패킷을 전송하고 다시 수신함으로써 대상 컴퓨터가 네트워크에 접속되어있는지를 확인하는 명령
- ping에 응답한다는 것은 접속 가능하다는 의미이며 이는 해킹이 가능하다는 의미로 해석될 수 있기 때문에 최근에는 방화벽을 통해 ping에 응답하지 않도록 설정하는 것이 일반적임

```
$ ping 접속대상주소
```

명령어가 동작하거나 대기중인 상태에서 **Ctrl+C**를 눌러 동작을 중단해야 함

### ping에 정상적으로 응답하는 경우

**Ctrl+C**를 눌러 ping을 종료하고 나면 5패킷을 보냈지만 5패킷을 수신했으므로 0% 손실되었다는 내용이 출력됨

```
leekh@my-server: ~$ ping itpaper.co.kr
PING itpaper.co.kr (211.249.62.211) 56(84) bytes of data.
64 bytes from 211.249.62.211 (211.249.62.211): icmp_seq=1 ttl=128 time=4.85 ms
64 bytes from 211.249.62.211 (211.249.62.211): icmp_seq=2 ttl=128 time=6.81 ms
64 bytes from 211.249.62.211 (211.249.62.211): icmp_seq=3 ttl=128 time=7.09 ms
64 bytes from 211.249.62.211 (211.249.62.211): icmp_seq=4 ttl=128 time=9.41 ms
64 bytes from 211.249.62.211 (211.249.62.211): icmp_seq=5 ttl=128 time=7.24 ms
^C
--- itpaper.co.kr ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 8256ms
rtt min/avg/max/mdev = 4.846/7.078/9.411/1.450 ms
leekh@my-server:~$
```

### ping에 응답이 없는 경우

**Ctrl+C**를 눌러 ping을 종료하고 나면 22패킷을 보냈지만 0패킷을 수신했으므로 100% 손실되었다는 내용이 출력됨

```
leekh@my-server: ~ -- 91x7$ ping naver.com
PING naver.com (223.130.200.104) 56(84) bytes of data.
^C
--- naver.com ping statistics ---
22 packets transmitted, 0 received, 100% packet loss, time 21503ms
leekh@my-server:~$
```

### 3) DNS 상태 점검 명령

- DNS 서버에 접속하여 동작이 잘 되는지 검사하는 명령
- DNS로부터 특정 도메인에 대한 IP주소를 조회함

```
$ nslookup 조회할도메인
```

```

leekh@my-server: ~
leekh@my-server:~$ nslookup itpaper.co.kr
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   itpaper.co.kr
Address: 211.249.62.211

leekh@my-server:~$ nslookup naver.com
Server:      127.0.0.53
Address:     127.0.0.53#53

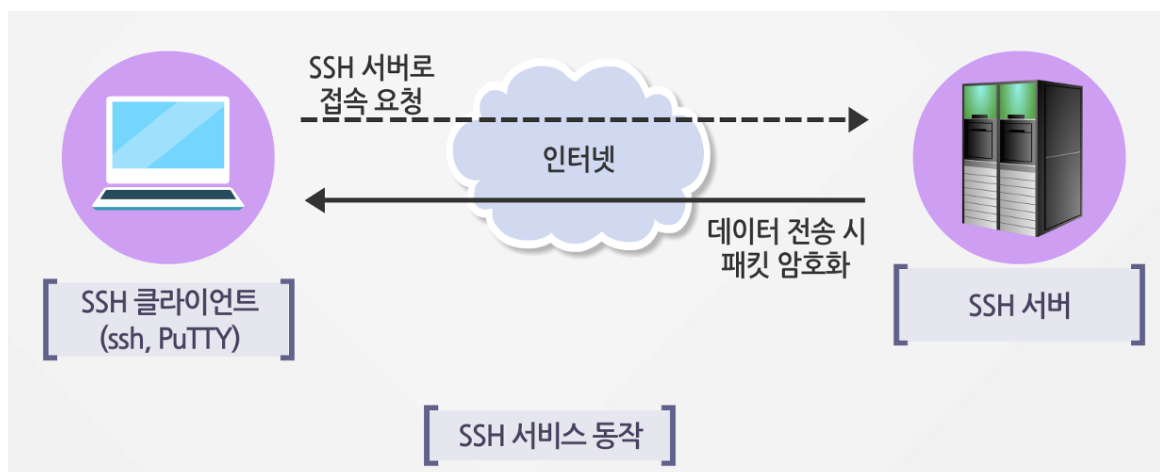
Non-authoritative answer:
Name:   naver.com
Address: 223.130.200.107
Name:   naver.com
Address: 223.130.195.200
Name:   naver.com
Address: 223.130.200.104
Name:   naver.com
Address: 223.130.195.95

leekh@my-server:~$

```

### #03. SSH 서비스

- 전송되는 데이터가 암호화 처리되는 보안이 강화된 원격 접속 서비스
- 과거에 사용되던 텔넷(Telnet) 방식의 경우 모든 데이터가 암호화 되지 않고 전송되어 악의적인 목적을 갖는 공격자가 중간에 패킷을 가로채어 분석 후 해킹을 시도할 수 있으므로 현재는 사용되지 않음 (ex: PC통신)



우분투 리눅스를 설치하는 과정에서 이미 SSH 서비스를 설치하여 사용하고 있음.

#### 1. SSH 서비스 동작 여부 확인하기

```
$ sudo systemctl status sshd
```

```
leekh@my-server: ~
leekh@my-server:~$ sudo systemctl status sshd
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-07-15 05:39:44 UTC; 3 days ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 753 (sshd)
     Tasks: 1 (limit: 2192)
    Memory: 6.6M
       CPU: 421ms
    CGroup: /system.slice/ssh.service
            └─753 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

lines 1-11...skipping...
```

## 2. 부팅시 자동 실행 여부 확인하기

```
$ sudo systemctl list-unit-files | grep sshd
```

```
leekh@my-server: ~
leekh@my-server:~$ sudo systemctl list-unit-files | grep sshd
sshd.service                                alias -
leekh@my-server:~$
```

## #04. 방화벽

- 미리 정의된 보안 규칙에 기반한, 들어오고 나가는 네트워크 트래픽을 모니터링하고 제어하는 네트워크 보안 시스템
- 방화벽 또는 파이어월이라고 함
- 일반적으로 신뢰할 수 있는 내부 네트워크, 신뢰할 수 없는 외부 네트워크 간의 장벽을 구성함
- 우분투 리눅스에서의 방화벽은 **ufw**라는 서비스가 담당하며 우분투 설치시 기본으로 설치됨
- 

### 1. 방화벽 시작 및 종료

#### 1) 방화벽 활성화

```
$ sudo ufw enable
```

#### 2) 방화벽 비활성화

```
$ sudo ufw disable
```

### 3) 방화벽 상태 확인

```
$ sudo ufw status
```

### 4) 특정 포트 접근 허용하기

ufw는 기본적으로 모든 port에 대해 외부 접근을 차단한다. ufw allow 명령어를 사용하면 서버의 특정 port로의 접근을 허용할 수 있다. 허용할 port의 범위를 지정할 수도 있다.

```
$ sudo allow [port_number]
$ sudo allow 2222 # 2222번 port 접근 허용
$ sudo allow 2222/tcp # 2222번 port tcp 접근 허용
$ sudo allow 3000:3010/tcp # 3000~3010 port tcp 접근 허용
```

### 5) 특정 IP 접근 허용하기

port 뿐만 아니라 ip를 통한 접근 권한 부여도 가능하다. 똑같이 ufw allow 명령어를 사용하여 해당 동작을 수행할 수 있다.

```
$ sudo ufw allow from [IP address]
$ sudo ufw allow from 123.4.56.7 # 123.4.56.7 ip에서 모든 port 접근 허용
```

### 6) 특정 IP 접근 제한하기

특정 IP로부터 공격이 들어오는 경우 ufw deny 명령어를 통해 특정 IP를 차단할 수 있다.

```
$ sudo ufw deny from [IP address]
$ sudo ufw deny from 123.45.6.78 # 123.45.6.78 IP에 대해 모든 port 접근 제한
```

### 7) 방화벽 규칙 삭제

ufw delete 명령어를 사용하여 ufw rule을 삭제할 수 있다.

```
$ sudo ufw delete [rule]
$ sudo ufw delete allow 2222/tcp # 2222/tcp 접근 허용 rule을 제거
```

### 8) 방화벽 규칙 다시 로드하기

```
$ sudo ufw reload
```

## 9) 방화벽 규칙 리셋하기

```
$ sudo ufw reset
```

## 10) ping 차단하기

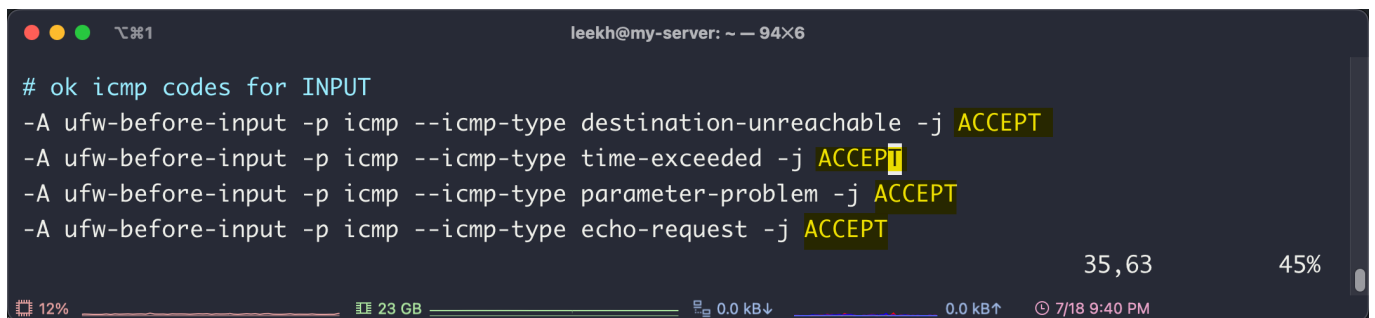
ufw는 기본적으로 ping 요청을 허용한다. 보안을 위해서는 ping 요청을 차단하는 것이 좋다.

`/etc/ufw/before.rules` 파일을 vi편집기로 연다.

```
$ sudo vi /etc/ufw/before.rules
```

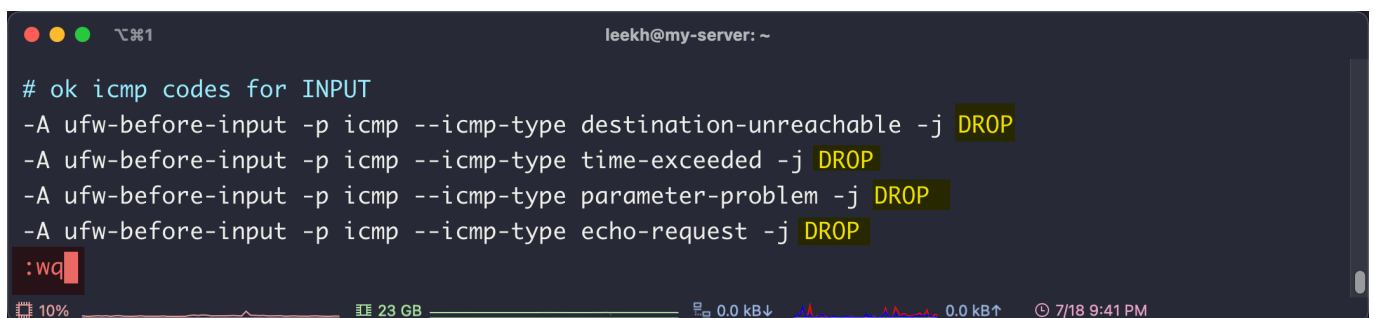
아래와 같은 부분을 찾아 맨 마지막의 **ACCEPT**를 **DROP**으로 변경한다.

변경전



```
leekh@my-server: ~ — 94x6
# ok icmp codes for INPUT
-A ufw-before-input -p icmp --icmp-type destination-unreachable -j ACCEPT
-A ufw-before-input -p icmp --icmp-type time-exceeded -j ACCEPT
-A ufw-before-input -p icmp --icmp-type parameter-problem -j ACCEPT
-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT
35,63 45%
```

변경후



```
leekh@my-server: ~
# ok icmp codes for INPUT
-A ufw-before-input -p icmp --icmp-type destination-unreachable -j DROP
-A ufw-before-input -p icmp --icmp-type time-exceeded -j DROP
-A ufw-before-input -p icmp --icmp-type parameter-problem -j DROP
-A ufw-before-input -p icmp --icmp-type echo-request -j DROP
:wq
10% 23 GB 0.0 kB↓ 0.0 kB↑ 7/18 9:41 PM
```

방화벽 재로드

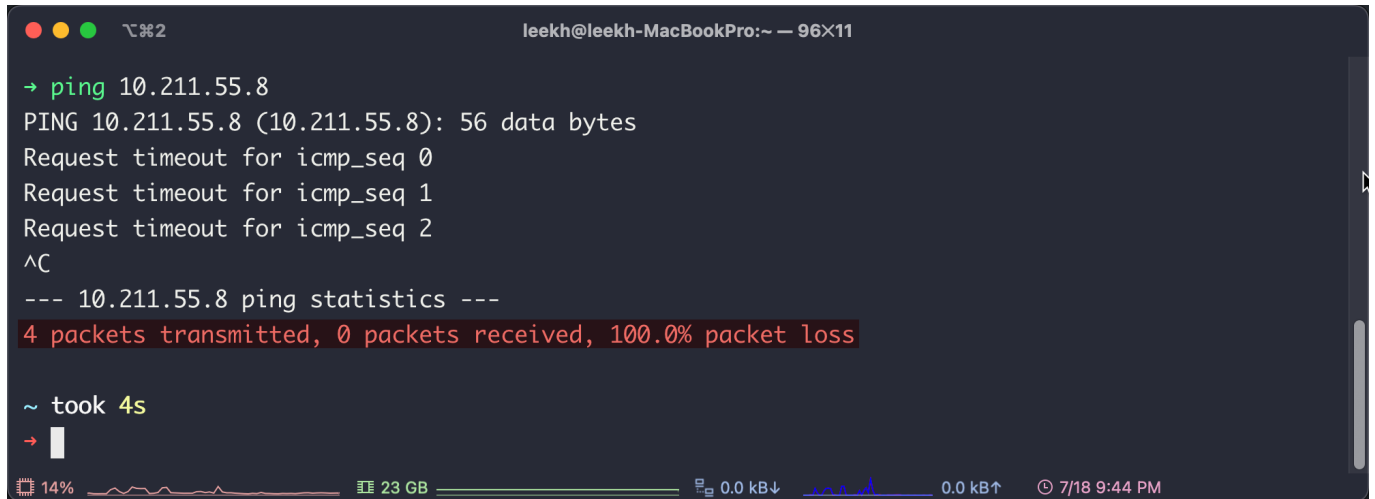
처리가 완료되면 방화벽을 다시 로드한다.

```
$ sudo ufw reload
```

결과확인



Mac, Window 등의 호스트 운영체제에서 Ubuntu로 ping접속을 시도하여 ping 응답이 없음을 확인한다.



```
leekh@leekh-MacBookPro:~ — 96x11
→ ping 10.211.55.8
PING 10.211.55.8 (10.211.55.8): 56 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
^C
--- 10.211.55.8 ping statistics ---
4 packets transmitted, 0 packets received, 100.0% packet loss

~ took 4s
→ █
```

The screenshot shows a terminal window on a Mac. The user has executed the command `ping 10.211.55.8`. The output shows that the ping failed with a 100.0% packet loss. The terminal window has a title bar with standard Mac window controls and a status bar at the bottom showing system information like battery level, memory usage, and network activity.