

# Linux 의 활용

## 목 차

1. 유닉스 시스템의 개요
2. 유닉스의 기본 사용
  3. 환경 설정
  4. 파일 편집기(vi)
5. 파일과 디렉토리 명령어
6. 프로세스 관리 명령어
  7. 인터넷 명령어
  8. 통신 명령어
  9. 쉘
10. 시스템 관리 명령어
11. 프로그램 설치 및 활용

## 1. 유닉스 시스템의 개요

### 1.1 유닉스의 특징

유닉스(UNIX)는 운영체제의 한 종류로써 다음과 같은 특징이 있다.

- 다중 사용자 처리(multi-user & multi process system)
- 시분할 시스템(Time sharing system)
- 계층적 파일 시스템(Hierarchical File system)
- 높은 생산성(High productivity)
- 업계 표준시스템(Industry standard system)
- 고수준 언어로 프로그래밍화(Programmed by High level language)
- 뛰어난 이식성(Good protability)
- 진보된 네트워킹 제공(Support high-level Networking facility)
- 보안시스템

### 1.2 유닉스의 구조

유닉스의 구조는 크게 다음 3 가지로 이루어져 있다.

#### 1.2.1 커널(kernel)

유닉스 시스템 구성의 기초를 이루는 핵심 부분으로서 소위 OS 라고 불리는 부분이다. 하드웨어와 사용자 프로그램의 중간에 위치하며 CPU, 메모리, 하드 디스크등의 하드웨어 자원을 제어하면서, 프로세스 스케줄링, 기억장치 관리, 파일 관리, 시스템 호출 인터페이스, Operator 콘솔 및 인터페이스 기능 멀티 프로그래밍 지원, 파일 시스템 서비스, 입출력 서비스 등의 기능을 사용자에게 제공한다. 명령을 해석하는 커맨드 인터프리터(interpreter)는 커널에 내장되어 있지 않으며 다른 사용자 프로그램과 똑같은 애플리케이션(application) 프로그램으로 독립되어 있다.

#### 1.2.2 셸(Shell)

사용자가 입력한 명령을 해석하고 실행하는 커맨드 인터프리터로서, 명령행의 해석, 바꿔 놓기, 입출력의 방향 변환 변수의 전개 등을 한다. 셸은 사용자와 커널간의 주고 받기를 하지만 다른 커맨드와 똑같은 실행 가능한 파일의 하나이며 커널 속에 내장되어 있는 것은 아니다. AT&T 버전, Berkley 버전 모두 본(Bourne) 셸이 표준으로 제공되고 있다. Bourne-셸과 Korn-셸의 기본 프롬프트는 \$ 이고 C-셸의 기본 프롬프트는 %이다. 시스템 관리자(super user)의 프롬프트는 #이다. 대표적인 셸은 다음과 같다.

- sh : Stephen Bourne 이 만든 셸로 모든 시스템에 들어있는

Bourne-셸이다.

- csh : sh 보다 기능이 보장되었으며 Bill Joy 가 만든 C-셸이다.
- ksh : sh 보다 기능이 보장되었으며 David Korn 이 만든 Korn-셸이다.
- bash: 기본적으로 들어 있지는 않고 소스를 구해서 자신의 시스템에 컴파일 하여야 하지만 위의 3 개와 비교할 때 가장 편리한 셸이라고 할 수 있다.(linux 에는 기본 탑재)

### 1.2.3 파일 시스템(File system)

유틸리티(utility), 애플리케이션(application) 및 데이터(data)를 저장하고 있다. 파일 시스템은 계층 구조로서 일반적인 트리 구조를 갖는다.

### 1.3 디렉토리 구조

디렉토리는 다른 파일들과 디렉토리에 관한 정보를 저장하는 논리적인 영역이다. 디렉토리는 특수한 형태의 데이터 파일로써 원하는 데이터를 자유롭게 읽을 수는 있지만 일반 사용자가 디렉토리를 수정하는 것은 불가능하다. 유닉스 시스템에서 가장 상위의 디렉토리는 유닉스 운영체제가 위치하고 있는 루트 디렉토리 즉 /로 표현된다. 루트 디렉토리 아래에는 특수 파일 및 일반 파일들이 위치한다.

- /bin : 시스템 구동 실행 파일들이 위치한다.
- /dev : 시스템의 모든 입출력 파일을 가지고 있다.
- /etc : 시스템 설정 파일, 프로그램, 유틸리티등이 있다.
- /home : 사용자의 홈 디렉토리이다.
- /lib : 공유 라이브러리가 위치한다.
- /usr : 실제 작업을 위한 응용 패키지들이 설치되어 있다.

## 2. 유닉스의 기본 사용

직접 워크스테이션(workstation)을 구동하거나 네트워크로 연결된 단말기 또는 일반 PC 를 이용하여 telnet 으로 유닉스 시스템에 접근한다. 유닉스 시스템에서는 대-소문자를 철저히 구분한다. 모든 명령어는 통상적으로 소문자로 입력한다.

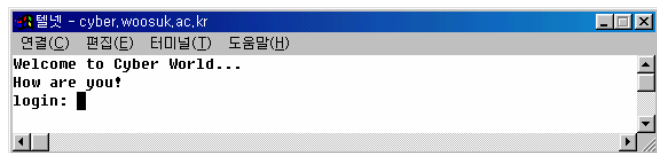
### 2.1 시스템의 시동

#### 2.1.1 접속하기

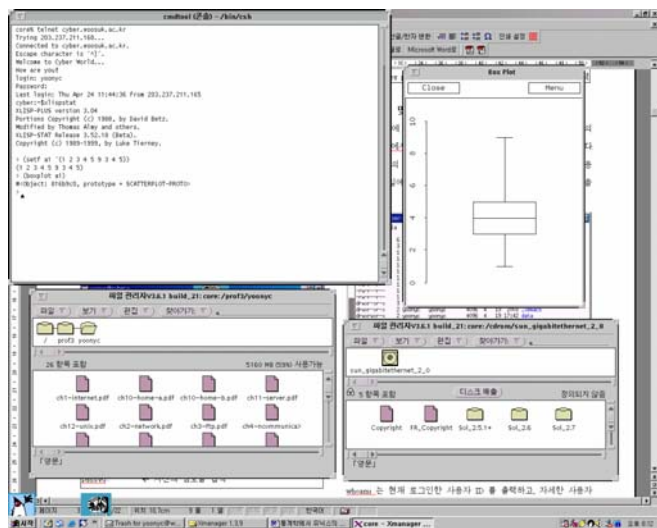
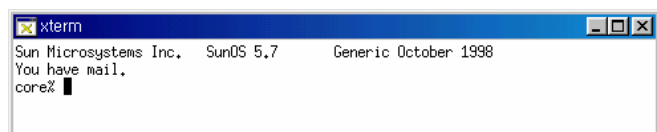
사용자의 PC 를 이용하여 원격지의 호스트에 접속하기 위해서는 정당한 계정(ID)과 비밀 번호(password)를 가지고 있어야 한다.

그리고 사전에 PC 의 네트워크 환경 설정을 해야 한다. 적절하게 설정이 끝나고 네트워크 관련 프로그램의 설치가 끝나면 원격지 호스트로의 접속 준비가 된 것이다.

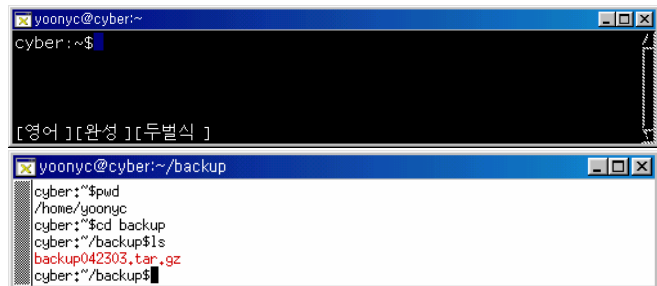
● 윈도우의 텔넷(telnet) 프로그램으로 접속하기



● Xmanager 프로그램으로 Sun 접속하기(xterm, openwindow)



● Xmanager 프로그램으로 Linux 접속하기



2.1.2 자신의 ID 와 암호를 입력하는 화면

(그림 12-4)의 접속 화면에서 사용자의 ID 와 암호를 입력하면 연결 된다.

login:	← 자신의 ID 를 입력
passwd:	← 자신의 암호를 입력

접근이 수락되면 화면에 다음과 같은 prompt 가 나온다.

\$	← Korn-셸, Bourne-셸의 기본 프롬프트
%	← C-셸의 기본 프롬프트
#	← 시스템관리자의 루트 권한 프롬프트

2.1.3 셸의 종류 확인

자신의 현재 사용하는 셸의 종류를 확인하려면 다음과 같이 입력한다. 그러면 셸의 종류와 셸의 구동 위치를 볼 수 있다.

```
$ echo $SHELL
/bin/bash
% echo $SHELL
/bin/csh
```

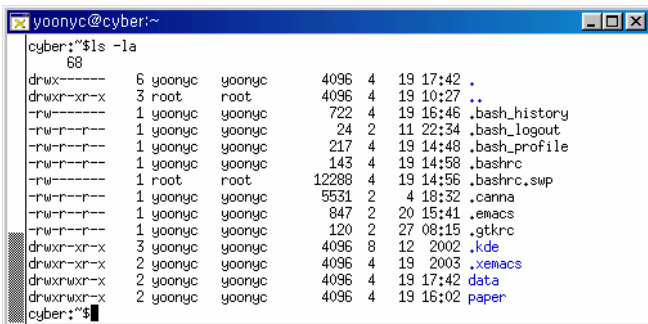
2.1.4 암호 변경(passwd)

암호를 바꾸고자 할 경우 사용한다.

```
$ passwd
passwd: Changing password for yoonyc
Enter login password: ← 원래 사용하던 암호를 입력
New password: ← 새롭게 변경시킬 암호를 입력
Re-enter new password:← 확인을 위해 변경될 암호를 다시 입력
```

2.1.5 환경 맞추기

셸의 종류에 따라 시스템에 login 할 때마다 유닉스는 사용자의 홈 디렉토리에서 .profile 파일들이나 .login 파일들을 먼저 조사한다. 파일이름 앞의 .은 일반적으로 보이지 않는 숨은 파일명이며 사용자는 이 파일에 적절한 자신의 환경 path 나 환경변수 설정 등을 할 수 있다.



2.1.6 종료

login 을 종료하려면 다음 명령을 입력한다.

```
$ exit
```

2.2 간단한 유닉스 명령어

2.2.1 who

현재 시스템을 사용중인 사람을 확인하고 싶을 때 사용한다. whoami 는 현재 로그인한 사용자 ID 를 출력하고, 자세한 사용자 정보는 who am i 와 같이 띄어쓰기를 하여 입력하면 된다.

```
cyber:~$who
root      :0          Apr 19 13:01
root      pts/1         Apr 19 13:01
root      pts/2         Apr 19 14:57
root      pts/0         Apr 19 13:13
yoonyc    pts/3         Apr 19 17:44 (203.237.211.165)
cyber:~$whoami
yoonyc
cyber:~$who am i
yoonyc    pts/3         Apr 19 17:44 (203.237.211.165)
```

### 2.2.2 finger

어떤 사람에 대하여 좀더 자세한 정보를 원할 때에 사용한다.

(예) [finger] [알고싶은 id]

```
cyber:~$finger yoonyc
Login: yoonyc                Name: YeoChang Yoon
Directory: /home/yoonyc      Shell: /bin/bash
On since Sat Apr 19 17:44 (KST) on pts/3 from 203.237.211.165
No mail.
No Plan.
```

### 2.2.3 bc

간단한 계산기이다.

```
cyber:~$bc
bc 1.06
Copyright 1991-1994, 1997, 1998, 2000 Free Software Foundation, Inc.
3+8 ← 엔터.
11 ← 계산 결과.
cyber:~$ ← [control] + d 를 누르면 계산기를 빠져 나온다.
```

### 2.2.4 cal

달력을 화면에 출력하는 명령이다. 다음의 예는 2003 년 4 월의 달력을 보여주고 있다.

```
$ cal 4 2003
    2003 년 4 월
일 월 화 수 목 금 토
    1  2  3  4  5
  6  7  8  9 10 11 12
 13 14 15 16 17 18 19
 20 21 22 23 24 25 26
 27 28 29 30
```

만약에 2003 년의 전체 달력을 보고 싶으면 다음과 같이 명령한다.

```
$ cal 2003
```

### 2.2.5 date

현재의 날짜와 시간을 화면에 나타내 준다.

```
cyber:~$date
2003. 04. 19. (토) 17:50:24 KST
```

## 3. 환경 설정

시스템에서 설정된 기본 셸의 종류에 따라서 시스템 로그인 시

에 초기 검색되는 파일들이 다르다. Bourne(Shell) 셸과 Korn(Korn) 셸 인 경우는 .profile 파일을, C 셸에서는 .login 파일과 .cshrc 파일을 먼저 참조하여 환경들을 설정하게 된다. 이 파일들에는 각각 사용자의 환경 변수들이 지정되어 있으며 Hidden(".dot")파일들로서 ls -la 명령으로 볼 수 있다.

### 3.1 env

현재 설정된 환경변수를 보여 준다.

```
cyber:~$env
REMOTEHOST=203.237.211.165
HOSTNAME=cyber.woosuk.ac.kr
PVM_RSH=/usr/bin/rsh
SHELL=/bin/bash
TERM=ansi
LESSCHARSET=ko
HISTSIZE=1000
QTDIR=/usr/lib/qt-3.1
USER=yoonyc
LS_COLORS=no=00:fi=00:di=01:34:ln=01:36:pi=40:33:so=01:35:bd=40:33:01:cd=40:33:01:or=01:05:37:41:mi=01:05:37:41:ex=01:32:*cmd=01:32:*exe=01:32:*com=01:32:*b
tm=01:32:*bat=01:32:*sh=01:32:*csh=01:32:*tar=01:31:*tgz=01:31:*arj=01:31:
*taz=01:31:*lzh=01:31:*zip=01:31:*z=01:31:*Z=01:31:*gz=01:31:*bz2=01:31:*
*bz=01:31:*tz=01:31:*rpm=01:31:*cpio=01:31:*jpg=01:35:*gif=01:35:*bmp=01:3
5:*x=01:35:*xpm=01:35:*png=01:35:*tif=01:35:
PVM_ROOT=/usr/share/pvm3
MAIL=/var/spool/mail/yoonyc
PATH=/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin:/home/yoonyc/
bin:/usr/local/xlisp-stat/bin
INPUTRC=/etc/inputrc
PWD=/home/yoonyc
LANG=ko_KR.eucKR
LAMHELPFILE=/etc/lam/lam-helpfile
PS1=hc:w$
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
SHLVL=1
HOME=/home/yoonyc
XPVM_ROOT=/usr/share/pvm3/xpvm
LOGNAME=yoonyc
LESSOPEN=/usr/bin/lesspipe.sh %s
G_BROKEN_FILENAMES=1
=/bin/env
```

### 3.2 PATH 지정

새로운 패스를 설정할 경우 기본 셸의 종류에 따라 .profile 나 .cshrc 파일을 다음과 같이 수정 한다. 다음의 예는 경로 /home/bin 을 새로 추가하는 경우이다.

- Bourne-셸, Korn-셸

```
$ vi .profile
PATH=.: /usr/bin:/home/bin
```

- C-셸

```
% vi .cshrc
set path=(. /usr/bin home/bin)
```

### 3.3 변경 내용 적용

변경 내용을 즉시 적용하려면 다음과 같다. 이 방법은 시스템을 재 부팅하거나, 로그 아웃한 다음에 다시 로그 인하지 않고 변경 내용을 시스템에 반영시킨다.

- Bourne-셸, Korn-셸

```
$ . .profile
```

- C 셸

```
% source .cshrc
```

- 프롬프트(bash) 모양 변경하기

--- bash script 에 사용가능한 옵션들 ---

\t 현재 시간을 HH:MM:SS 형식으로 보여준다.

\d 날짜를 "요일 월 일" 형식으로 보여준다.

\s 셸의 이름을 보여준다.

\w 현재 작업디렉토리를 완전경로로 보여준다.

\W 현재 작업디렉토리의 이름을 보여준다.

\u 현재 사용자의 사용자명

\h 호스트이름

!\ 이 명령의 히스토리 번호

\nnn 8 진수 nnn 에 해당하는 문자

[ 비출력문자의 시퀀스를 시작한다.

] 비출력문자의 시퀀스를 마친다.

(예제) PROMPT 설정변경

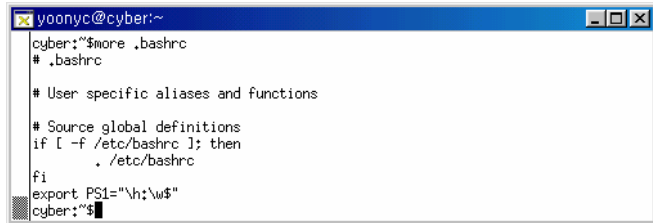
[.bashrc] 파일에 다음의 내용을 추가한다.

```
export PS1="\h:\w#"
```

[결과 1]

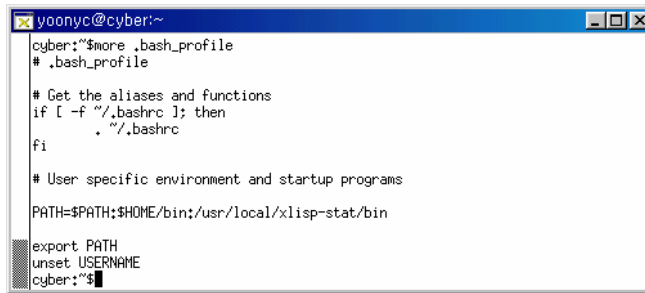
```
[cyber:/home/hongyk]# <-- 이와같은 프롬프트가 만들어짐
```

[결과 2]



```
[cyber:/home/hongyk]# <-- 이와같은 프롬프트가 만들어짐
```

- 프로그램 경로 지정



셸을 반영시키려면,

\$.bash\_profile 또는 logout → login 한다.

csh 인 경우는,

```
%source .cshrc
```

### 3.4 그 외의 환경 설정

- alias

별명을 주는 명령이다. 예를들어 .cshrc 나 .profile 파일의 내부에 C-셸인 경우는 alias dir 'ls -la' 그리고 korn-셸인 경우는 alias dir='ls -la'라고 입력하여 놓으면 사용자가 dir 을 실행할 때 항상 ls -la 가 실행되어 파일의 목록을 살펴볼 수 있게 된다.

- history

사용자가 사용한 명령어를 기억하여 재입력하는 불편함을 덜어 준다. 히스토리 기능을 사용하기 위해서는 셸의 종류에 따라 초기 설정 파일에 다음과 같이 설정한다. 만약에 사용한 명령어를 100 개까지 기억하여 볼 수 있도록 설정하기 위해서는 C-셸인 경우에 .cshrc 파일의 내부에 set history=100 을 삽입하고, Bourne-셸, Korn-셸인 경우에는 .profile 파일의 내부에 HISTORY=100 을 삽입한다.

C-셸에서 사용 예는 다음과 같다(bash 도 같다). 지금까지 사용한 명령어를 보기 위하여 history 를 실행해 보면 다음과 같다.

```
% history
7 source .cshrc
8 ls
9 cd
10 history
11 cd /home/yoonyc
12 ls
13 gzip -cd test.tar.gz | tar xvf -
14 ls
15 history
```

여기서 11 번을 다시 실행하고 싶을 때에는 다음과 같다.

```
% !11
```

Korn-셸에서 사용 예는 다음과 같다. 지금까지 사용한 명령어를 보기 위하여 history 를 실행해 보면 다음과 같다.

```
$ history
7 source .cshrc
8 ls
9 cd
10 history
11 cd /home/yoonyc
12 ls
13 gzip -cd test.tar.gz | tar xvf -
14 ls
15 history
```

여기서 11 번을 다시 실행하고 싶을 때에는 다음과 같다.

```
$ r11
```

Bourne-셸(bash)일 경우에는 ↑키를 누를 때마다 지금까지 실행한 명령어가 최근부터 차례로 나오게 된다.

● prompt

프롬프트 모양을 정의 한다.

Bourne-셸, Korn-셸인 경우에는 .profile 파일의 내부에 다음 명령을 삽입한다.

```
PS1=": "
PS1="hostname": "
```

C-셸인 경우에는 .cshrc 파일의 내부에 다음 명령을 삽입한다.

```
set prompt="% "
set prompt="hostname"!:
```

● umask

사용자가 파일을 기록할 때에 허가 모드(permission)를 지정하여 준다. umask 는 chmod 와는 다르게 제한 하고자 하는 코드를 입력하면 된다.

(예) umask 023

사용자는 제한하는 것이 없이(0) rwx 을 주고, 그룹은 쓰기(2)를 제한하며, 기타 사용자는 쓰기+실행=3 을 제한 한다. 그러므로 사용자가 작성하는 모든 파일의 허가 모드는 다음과 같이 생성된다.

```
rwxr-xr--
```

### 3.5 일반적인 환경설정의 예

UNIX 사용자의 일반적인 환경설정을 위해서는 다음과 같은 .cshrc 파일을 자신의 계정에 만들어 주어야 한다.

```
# @(#)cshrc 1.11 98/12/29 SMI
umask 022
set path=(/bin /usr/bin /usr/ucb /usr/sbin /etc .)
if ( $?prompt ) then
    set history=20
endif
set prompt = "core# pwd"> "
alias cd 'cd !*; set prompt = "[core# pwd`]"
alias dir "ls -la |more"
alias cls clear
alias h history
alias netscape "/home1/netscape/netscape"
alias mail /usr/ucb/mail
stty erase ^H
setenv LANG ko
```

## 4. 파일 편집기(vi)

환경 설정을 고치기 위해서는 기존의 파일을 수정해야 하는데, 이때 간단하게 사용할 수 있는 편집기가 vi 이다. 다음의 예는 sample 라는 파일을 만들어 편집하는 경우이다. 먼저 다음과 같이 입력한다.

```
% vi sample
```

그러면 화면이 다음과 같이 나타난다.

```
~
~
~
~
"sample" [New file]
```

새로운 내용을 입력하기 위해서는 'esc'키를 누른 후 'i'자를 누른다. 그러면 이제부터 글자가 입력될 수 있다. 다음의 예처럼 입력하자. 참고로 키보드의 화살표 키는 사용할 수 없다.

```
동해물과 백두산이 마르고 닳도록
하나님이 보우하사 우리나라 만세
무궁화 삼천리 화려강산
대한사람 대한으로 길이 보전하세
```

입력 후 파일을 저장하고 나오려면 'esc'키를 누른 후 ':'키를 누른다. 그러면 화면의 좌측 하단에 다음과 같이 :가 나타난다. : 옆에 wq!를 입력하고 'enter'키를 누르면 파일 sample 이 저장된 후에 빠져나오게 된다.

```
동해물과 백두산이 마르고 닳도록
하나님이 보우하사 우리나라 만세
무궁화 삼천리 화려강산
대한사람 대한으로 길이 보전하세
~
~
:wq!
```

### 4.1 명령어 모드

vi 편집기에서 'esc' 키를 한 번 누르면 입력 상태에서 명령어 모드로 변환이 된다. 명령어 모드에서는 다음과 같은 명령어들을 사용할 수 있다.

● 주요 명령어

l	커서를 오른쪽으로 이동
h	커서를 왼쪽으로 이동
j	커서를 아래로 이동
k	커서를 위로 이동
[control] + f	Page Down
[control] + b	Page Up
H	화면의 첫 라인 첫 컬럼으로 이동
M	화면의 중간 라인 첫 컬럼으로 이동
L	화면의 끝 라인 첫 컬럼으로 이동
w	오른쪽 단어로 이동
b	왼쪽 단어로 이동
^	현재 라인의 첫 문자로 이동
\$	현재 라인의 마지막 문자로 이동
a	커서 뒤에 입력
i	커서 앞에 삽입
r	커서 위치의 문자를 다른 문자로 수정
u	바로 전에 행한 한번의 명령을 취소
U	현재 라인에서 했던 모든 명령을 취소
cw	현재 커서에 있는 단어를 수정
cc	현재 커서에 있는 라인을 모두 수정
dw	커서 위치에서 오른쪽 방향으로 한 단어를 삭제
db	커서 위치에서 왼쪽 방향으로 한 단어를 삭제
dd	커서가 위치한 현재 라인을 삭제
/문자열	문자열을 순 방향으로 검색
?문자열	문자열을 역 방향으로 검색
yy	현재의 라인을 버퍼에 복사
p	현재 커서의 오른쪽 또는 아래 라인에 버퍼 내용 붙이기

● 마지막 라인 모드

명령어 모드에서 다시 : 을 누르면 마지막 라인 모드가 된다.

마지막 라인 모드에서는 아래와 같은 명령어를 사용할 수 있다.

:w	파일 저장
:q	저장 안하고 빠져 나옴
:ZZ	수정한 파일을 저장 후 빠져 나옴
:wq!	수정한 파일을 강제로 저장 후 빠져 나옴
:w [파일이름]	현재의 내용을 [파일이름]으로 저장한다
:r [파일이름]	외부 파일 읽어 들이기
!:	잠시 셸로 나가기
:n	n 번째 라인으로 이동

5. 파일과 디렉토리 명령어

5.1 ls

디렉토리 안의 내용을 열람하기 위한 명령어이다.

```
cyber:~$ls
data paper
```

ls -al 을 하면 좀더 자세한 정보를 보여 준다.

```
cyber:~$ls -la
합계 68
drwx----- 6 yoonyc yoonyc 4096 4월 19 17:42 .
drwxr-xr-x 3 root root 4096 4월 19 10:27 ..
-rw----- 1 yoonyc yoonyc 841 4월 19 17:43 .bash_history
-rw-r--r-- 1 yoonyc yoonyc 24 2월 11 22:34 .bash_logout
-rw-r--r-- 1 yoonyc yoonyc 217 4월 19 14:48 .bash_profile
-rw-r--r-- 1 yoonyc yoonyc 143 4월 19 14:58 .bashrc
-rw----- 1 root root 12288 4월 19 14:56 .bashrc.swp
-rw-r--r-- 1 yoonyc yoonyc 5531 2월 4 18:32 .canna
-rw-r--r-- 1 yoonyc yoonyc 847 2월 20 15:41 .emacs
-rw-r--r-- 1 yoonyc yoonyc 120 2월 27 08:15 .gtkrc
drwxr-xr-x 3 yoonyc yoonyc 4096 8월 12 2002 .kde
drwxr-xr-x 2 yoonyc yoonyc 4096 4월 19 2003 .xemacs
drwxrwxr-x 2 yoonyc yoonyc 4096 4월 19 17:42 data
drwxrwxr-x 2 yoonyc yoonyc 4096 4월 19 16:02 paper
```

-R 은 서브디렉토리의 내용까지 출력 한다.

-t 은 시간 순서로 정렬한다.

-F 은 디렉토리, 실행파일, 텍스트 파일을 구분하여 나열한다.

5.2 cat

파일의 내용을 보는 명령어이다. .bashrc 의 내용을 보고 싶다면

cat .bashrc 이라고 입력한다.

```
cyber:~$cat .bashrc
# .bashrc
# User specific aliases and functions
# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi
export PS1="h:w$"
```

cat 명령은 파일들을 합치는 데도 이용할 수 있다.

```
cyber:~$ cat fileA fileB > fileC
```

즉 fileA 와 fileB 를 합쳐서 fileC 를 만든다. 만일 fileC 가 원래 있었다면 fileC 의 원래 내용은 삭제 된다.

다음의 예는 fileB 의 원래 내용 밑에 fileA 의 내용을 추가한다.

여기서 >는 출력, >>는 덧붙임, <는 입력을 의미 한다.

```
cyber:~$ cat fileA >> fileB
```

만약 원하는 내용을 love.txt 란 파일로 보내려면, 파일 생성 명령어 cat 을 입력후, 내용을 love.txt 라는 곳으로 보내라는 리디렉션(redirection) 기호 '>'를 입력한다. 그 다음에 원하는 내용을 입력하면 된다. 그리고 엔터키를 입력후 ,Ctrl+z 한다.

입력한 내용은 \$ cat life.txt 으로 확인된다.

```
cyber:~$ cat > love.txt
I love you!

[1]+ Stopped          cat >love.txt
cyber:~$ cat < love.txt
I love you!
```

### 5.3 pwd

현재 디렉토리의 위치를 알려 준다.

```
cyber:~$pwd
/home/yoonyc
```

### 5.4 cd

디렉토리를 변경 한다. 여기서 ..은 상위 디렉토리를 의미 한다.

```
$ cd gdhong ← gdhong 이라는 디렉토리로 이동 한다.
$ cd ← 사용자의 홈디렉토리로 이동 한다.
$ cd .. ← 바로 위 디렉토리로 이동 한다
```

### 5.5 mkdir

디렉토리를 새로 만든다.

```
$ mkdir gdhong ← gdhong 디렉토리를 만든다.
```

### 5.6 rmdir

디렉토리를 삭제 한다.

```
$ rmdir gdhong ← gdhong 디렉토리를 삭제한다.
```

### 5.7 mv

파일을 이동 한다. fileA 는 사라지고 fileA 가 fileB 로 바뀐다.

```
$ mv fileA fileB ← fileA 를 fileB 로 이동한다
```

### 5.8 cp

파일을 복사 한다. fileA 는 그대로 남고 fileA 가 fileB 로 복사된 다.

```
$ cp fileA fileB ← fileA 를 fileB 로 복사 한다.
```

-r 은 하위 디렉토리의 파일을 같이 복사 한다.

```
$ cp -r dirA dirB ← dirA 의 모든 파일과 서브디렉토리를 dirB 로 복사 한다.
```

### 5.9 rm

파일을 삭제 한다.

```
$ rm fileA ← fileA 를 삭제 합니다.
```

-r 은 하위 디렉토리의 파일을 같이 삭제 한다.

```
$ rm -r dirA ← dirA 의 모든 파일과 서브 디렉토리를 삭제 한다.
```

-i 는 확인 절차가 추가 된다. 즉 삭제할 파일을 제시하고 사용자에게 확인하게 한다.

```
$ rm -i dirA
```

### 5.10 링크

링크(Link)란 하나의 파일이나 디렉토리에 다른 이름을 연결해 준다.

(예) [ln -s] [존재하는원파일이름] [새로만들다른파일이름]

여기서 s 는 symbloic link 를 뜻한다.

```
$ ls -al
-rw-rw-r-- 1 yoonyc user 6 12 20 15:33 my-file
```

현재 my-file 만 있다. 이제 이미 존재하는 my-file 을 다른 이름인 your-file 을 새로 만들어 연결 한다. 확인한 결과는 다음과 같다.

```
$ ln -s my-file your-file
$ ls -al
-rw-rw-r-- 1 yoonyc user 6 12 20 15:33 my
lrwxrwxrwx 1 yoonyc user 2 12 20 15:33 you -> my
```

다음의 예는 현재 있는 디렉토리 /home/netscape 를 새로만든 디렉토리인 /usr/local/netscape 로 연결한다.

```
$ ln -s /home/netscape /usr/local/netscape
```

### 5.11 diff

파일내용의 차이점을 file3 에 출력한다.

```
$ diff file1 file2 > file3
```

### 5.12 tail

파일의 뒷부분을 출력한다. 예를 들어 last 기록의 뒷부분 5 줄만 보고 싶을 경우는 다음과 같다.

```
$ last | tail -5
reboot system boot Tue Oct 22 17:39
root console Tue Oct 22 17:24 - down (00:15)
reboot system boot Tue Oct 22 17:22
```

### 5.13 more

파일의 내용을 한 화면씩 출력한다.

```
$ last | more
```

### 5.14 wc



지정한 파일에서 단어수, 줄수, 문자수 등을 알아볼 수 있다. 다음의 예는 .profile 파일에 15 줄이 있다는 것을 보여주고 있다.

```
$ wc -l .profile
15 .profile
```

### 5.15 grep

파일들의 내용에서 특정 단어를 찾고자 할 경우 사용한다. 아래 예는 /etc/passwd 파일에서 yoonyc 가 들어있는 행을 찾아 준다.

```
$ grep yoonyc /etc/passwd
yoonyc:x:203:200::/home/yoonyc:/bin/bash
$ ls | grep rm ← 주어진 패턴을 포함하는 파일을 찾는다.
rm.txt
rmdir.txt
```

### 5.16 find

특정 파일을 찾고자 할 경우 사용 한다. 아래 예는 루트(/)에서 부터 찾는데, 하드전체에서 이름(-name)이 passwd 인 파일을 찾아 화면에 보여달라(-print)는 명령이다.

```
$ find / -name passwd -print
/usr/bin/passwd
/export/home/ftphome/etc/passwd
/var/adm/passwd
/etc/default/passwd
/etc/passwd
```

현재 디렉토리에서 그 이하의 파일을 찾기 위한 명령은 다음과 같다.

```
$ find -name netscape -print
```

### 5.17 whereis

실제 프로그램이 어떤 디렉토리에 존재하는지 관련된 모든 경로로명을 보여줌

```
cyber:~$whereis vi
vi: /bin/vi /usr/share/vi /usr/share/man/man1/vi.1.gz
```

### 5.18 man

주어진 명령어나 유틸리티의 매뉴얼을 출력한다.

```
$ man find
```

종료할 때에는 'q'를 입력한다.

## 5.19 파일의 압축 및 복원

### 5.19.1 tar

여러 개의 파일이나 디렉토리를 하나의 파일로 묶어 주거나 테이프 등에 백업을 한다. 다음의 예는 tmp, work 디렉토리내의 모든 파일을 모두 backup.tar 파일로 묶어주는 예이다.

```
$ tar cvf backup.tar tmp work
```

풀 때는 다음과 같다.

```
$ tar xvf backup.tar
```

### 5.19.2 compress

파일을 압축한다. 위의 backup.tar 를 압축하고 결과를 보면 다음과 같다.

```
$ compress backup.tar
$ ls
backup.tar.Z
```

압축을 해제할 때는 다음과 같다.

```
$ uncompress backup.tar.Z
```

### 5.19.3 gzip

파일을 압축한다.

```
$ gzip -9 backup.tar
```

압축을 해제한다.

```
$ gzip -d backup.tar.gz
```

파일의 확장자가 .tgz 인 경우도 다음과 같다.

```
$ gzip -d backup.tgz
```

만약에 backup.tar.gz 의 압축을 풀고 동시에 묶인 파일을 풀어주는 방법은 다음과 같다.

```
$ gzip -cd backup.tar.gz | tar xvf -
```

## 5.20 C 프로그램의 컴파일 및 실행

### 5.20.1 gcc

gcc 는 GNU C 컴파일러로서 가장 용도가 다양하고 진보된 컴파일러 중의 하나이다. 대부분의 C 컴파일러(cc)와는 다르게 gcc 는 ANSI C 표준뿐만 아니라 C++, Object-oriented C 등을 지원하며 일반 UNIX 프로그래밍 환경에서 C/C++프로그래밍에 매우 효과적인 환경을 제공해 준다.

먼저 간단한 C/C++프로그램을 다음과 같이 작성한다.

```
#include <stdio.h> /* 파일이름=hello.c */
int main()
{
    printf("Hello, World!\n");
    return 0; /* Just to be nice */
}
```

gcc 를 이용하여 컴파일 한 후 실행시키면 다음과 같다. 여기서 gcc hello.c 는 a.out 이라는 실행파일을 생성시킨다.

```
$ gcc hello.c
$ ls -la
total 7
drwxr-xr-x  2 yoonyc  users  1024 Jan 10 13:59 .
drwx----- 13 yoonyc  users  1024 Jan 10 13:08 ..
-rwxr-xr-x  1 yoonyc  users  3928 Jan 10 13:59 a.out
-rw-r--r--  1 yoonyc  users   97 Jan 10 13:44 hello.c
$ ./a.out
Hello, World!
```

만약에 컴파일 결과의 실행파일 이름을 hello.out 으로 만들고 싶다면 다음과 같이 한다.

```
$ gcc -o hello.out hello.c
$ ls -la
total 11
drwxr-xr-x  2 yoonyc  users  1024 Jan 10 14:07 .
drwx----- 13 yoonyc  users  1024 Jan 10 13:08 ..
-rwxr-xr-x  1 yoonyc  users  3928 Jan 10 13:59 a.out
-rw-r--r--  1 yoonyc  users   97 Jan 10 13:44 hello.c
-rwxr-xr-x  1 yoonyc  users  3928 Jan 10 14:07 hello.out
$ ./hello.out
Hello, World!
```

여러 개의 C 프로그램(foo.c, bar.c, chart.c)을 하나의 실행파일(start)로 컴파일하고 싶다면 다음과 같다.

```
$ gcc -o start foo.c bar.c chart.c
$ ls -la
total 17
drwxr-xr-x  2 yoonyc  users  1024 Jan 10 14:07 .
drwx----- 13 yoonyc  users  1024 Jan 10 13:08 ..
-rwxr-xr-x  1 yoonyc  users  3928 Jan 10 13:59 a.out
-rw-r--r--  1 yoonyc  users   342 Jan 10 14:08 bar.c
-rwxr-xr-x  1 yoonyc  users  7129 Jan 10 14:23 start
-rw-r--r--  1 yoonyc  users   866 Jan 10 14:11 chart.c
-rw-r--r--  1 yoonyc  users   902 Jan 10 14:15 foo.c
-rw-r--r--  1 yoonyc  users   97 Jan 10 13:44 hello.c
-rwxr-xr-x  1 yoonyc  users  3928 Jan 10 14:07 hello.out
$ ./start
```

만약에 원시 프로그램(foo.c)의 Object 파일(foo.o)만을 생성시킨다면 다음과 같다.

```
$ gcc -c foo.c
$ ls -la
total 14
drwxr-xr-x  2 yoonyc  users  1024 Jan 10 14:07 .
drwx----- 13 yoonyc  users  1024 Jan 10 13:08 ..
-rw-r--r--  1 yoonyc  users   342 Jan 10 14:08 bar.c
-rwxr-xr-x  1 yoonyc  users  7129 Jan 10 14:23 start
-rw-r--r--  1 yoonyc  users   866 Jan 10 14:11 chart.c
-rw-r--r--  1 yoonyc  users   902 Jan 10 14:15 foo.c
-rw-r--r--  1 yoonyc  users   902 Jan 10 14:30 foo.o
```

따라서 다음과 같이 실행한 결과 파일들(start, start1)은 동일한

실행파일이다.

```
$ gcc -o start1 foo.c bar.c chart.c
$ gcc -c foo.c
$ gcc -c bar.c
$ gcc -c chart.c
$ gcc -o start1 foo.o bar.o chart.o
```

### 5.20.2 cc

cc 는 UNIX C 컴파일러 명령으로서 앞의 gcc 명령과 유사한 기능을 한다.

```
%cc -o hello.out hello.c
%ls -la
total 20
drwxr-xr-x  2 yoonyc user  512 1월 10일 14:35 .
drwxr-xr-x 13 yoonyc user  512 1월 10일 14:33 ..
-rw-r--r--  1 yoonyc user   97 1월 10일 14:34 hello.c
-rwxr-xr-x  1 yoonyc user 6160 1월 10일 14:35 hello.out
%hello.out
Hello, World!
```

### 5.21 패키지 관리

RPM 이란 RedHat Package Manager 의 약칭으로, 바이너리, 설정파일, 라이브러리, 다크멘트 등을 일괄 관리하는 툴이다. 인텔 프로세서용 패키지를 예로 들면, 마지막에 i386.rpm 이라는 확장자가 붙기 때문에 금방 알 수 있을 것이다.

RPM 은 아래와 같은 특징들이 있다.

- 인스톨, 업그레이드, 언인스톨이 용이하다.
- 데이터베이스를 검색해서 패키지나 특정 파일을 찾아내는 기능을 가지고 있다.
- 패키지에 필요한 파일, 의존성 등의 유무 등을 조사하는 기능을 갖고 있다.

설치되어 있는 패키지 확인

```
root:~#rpm -qa | grep ftp
ftp-0.17-17
gftp-2.0.14-2
ncftp-3.1.5-4
vsftpd-1.1.3-8
lftp-2.6.3-3
```

특정 패키지 삭제

```
root:~#rpm -e lftp-2.6.3-3
root:~#rpm -e lftp ← 파일 이름만 써도 된다
```

특정 패키지 설치

```
root:~#rpm -Uvh lftp-2.6.3-3
```

## 6. 프로세스 관리 명령어

### 6.1 ps

현재 활동중인 프로세스를 출력한다.

```
cyber:~$ps
  PID TTY          TIME CMD
 7635 pts/3    00:00:00 bash
 7949 pts/3    00:00:00 ps
cyber:~$ps -ef | grep telnetd
root  7633 2356  0 17:43 ?        00:00:00 in.telnetd: 203.237.211.165
yoonyc 8110 7635  0 20:58 pts/3  00:00:00 grep telnetd
```

### 6.2 last

사용자와 터미널에 대한 로그인 및 로그아웃 정보를 보여준다.

```
$ last
yoon  console  :0  Tue Oct 22 18:04 - 18:30 (00:25)
planet console  :0  Tue Oct 22 18:02 - 18:04 (00:01)
askim  console  :0  Tue Oct 22 18:00 - 18:02 (00:01)
yoon   console  :0  Tue Oct 22 17:57 - 18:00 (00:02)
reboot system boot  Tue Oct 22 17:56
root   console  :0  Tue Oct 22 17:40 - 17:54 (00:14)
```

### 6.3 head

파일의 앞부분을 출력한다. 예를 들어 last 기록의 앞부분 10 줄만 보고 싶을 경우 다음과 같다. 여기서 | 는 왼쪽의 실행 결과를 오른쪽으로 전달한다.

```
$ last | head -10
ftp  ftp  cyber  Thu Dec  5 09:35
yoon console :0  Thu Dec  5 09:30 - 11:45 (02:15)
yoon ftp  cyber  Wed Dec  4 21:00 - 21:02 (00:01)
yoon console :0  Wed Dec  4 20:53 - 21:02 (00:09)
ftp  ftp  cyber  Wed Dec  4 20:04 - down (00:47)
yoon ftp  cyber  Wed Dec  4 17:56 - 17:57 (00:00)
ftp  ftp  cyber  Wed Dec  4 17:10 - 17:56 (00:46)
Broken Pipe
```

### 6.4 su

현재 상태에서 다른 사용자로 다시 로그인 한다.

```
$ su yoon
```

yoon 으로 로그인 한 후에 다시 원래 상태로 돌아가는 명령은 다음과 같다.

```
$ exit
```

**\$ su** ← 직접 root 유저로서 로그인했을 때와는 달리 현재 유저의 환경을 그대로 물려받은 채 root 의 권한을 가진다. 이로 인해 PATH 의 설정도 원래의 유저와 동일하게 되어 있으므로 관리자 커맨드가 놓여있는 /sbin 이나 /usr/sbin 등은 PATH 에 포함되어 있지 않게 된다.

**\$ su -** ← 직접 root 유저로서 로그인했을 때와 동일한 환경으로 초기화가 되므로 /sbin 이나 /usr/sbin 등도 패스에 들어있게 된다

### 6.5 df

파일 시스템별 하드의 용량을 나타내 준다. df -k 는 디스크의 저장 상태와 가용공간등을 보여준다.

```
% df
/                (/dev/dsk/c0t0d0s0): 289552 블럭   92810 파일
/usr             (/dev/dsk/c0t0d0s6):1221628 블럭  465838 파일
/proc           (/proc                ):      0 블럭   8024 파일
/dev/fd         (fd                  ):      0 블럭    0 파일
/var            (/dev/dsk/c0t0d0s3): 147334 블럭  181297 파일
/usr/local     (/dev/dsk/c0t0d0s7):1751496 블럭  957710 파일
/usr2           (/dev/md/dsk/d0      ):5818082 블럭 2485693 파일
/tmp            (swap                ):1839424 블럭   94487 파일

% df -k
파일시스템      K 바이트  사용  가용  용량  설치지점
/dev/dsk/c0t0d0s0 192799  48023 125506 28%  /
/dev/dsk/c0t0d0s6 962983 352169 514524 41%  /usr
/proc              0         0      0      0%  /proc
fd                  0         0      0      0%  /dev/fd
/dev/dsk/c0t0d0s3 361807 288140 37487  89%  /var
/dev/dsk/c0t0d0s7 1927014 1051266 683048 61%  /usr/
/dev/md/dsk/d0     4990323 2081282 2410011 47%  /usr2
swap                920224    16  920208  1%  /tmp
```

### 6.6 du

현재 디렉토리와 포함된 서브 디렉토리의 사용 양을 알려 준다. 여기서는 전체 98,968KB 를 사용하고 있다.

```
% du
2      ./elm
30     ./Mail
2      ./fm
1186   ./comstat1
95026  ./comstat
2602   ./yoon
58     ./exam/exam01-2
60     ./exam
2      ./test
98968
```

### 6.7 stty

단말기의 설정 상황을 확인 하거나 변경할 수 있다. 다음의 예는 현재의 단말기 상황을 보여 준다.

```
% stty -a
speed 9600 baud; 25 rows; 80 columns;
intr = ^C; quit = ^\; erase = ^H; kill = ^U; eof = ^D; eol = ^@
eol2 = ^@; start = ^Q; stop = ^S; susp = ^Z; dsusp = ^Y; reprint = ^R
discard = ^O; werase = ^W; lnext = ^V
-parenb -parodd cs8 -cstopb hupcl cread -clocal -parext
-ignbrk brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl -iuclc
ixon -ixany ixoff imaxbel
isig icanon -xcase echo echoe echok -echonl -noflsh
-tostop echoctl -echoprtr echoke -flusho pending iexten
opost -olcuc onlcr -ocrnl -onocr -onlret -ofill -ofdel
```

### 6.8 free

스왑 용량과 메모리 정보를 출력한다.

```
cyber:~$free
              total    used    free  shared  buffers  cached
Mem:          513300  502632  10668    0      37392  320568
-/+ buffers/cache: 144672  368628
Swap:         1044216  11128   1033088
```

```
$ nslookup cyber.woosuk.ac.kr
Server: core.woosuk.ac.kr
Address: 210.93.6.2

Non-authoritative answer:
Name:   cyber.woosuk.ac.kr
Address: 203.237.211.168
```

### 6.9. CPU 를 사용하는 Process 죽이기

```
% ps ← 현재의 프로세서 번호를 알 수 있는 명령
PID TTY TIME CMD
1034 pts/1 0:00 csh
% kill -9 1034 ← 프로세서 번호 1034 를 죽인다.
```

### 7.3 finger

인터넷 호스트에 있는 사용자들에 대한 공개정보를 제공한다.

```
$ finger yoonyc@css.woosuk.ac.kr
[cyber.woosuk.ac.kr]
Login: yoonyc Name: Yoon YeoChang
Directory: /home/yoonyc Shell: /bin/bash
Office: Woosuk
On since Wed Dec 30 09:40 (KDT) on tty9 from cyber
 1 hour 17 minutes idle
Last login Wed Dec 30 10:21 (KDT) on tty6 from 203.237.211.165
Mail last read Wed Dec 16 21:26 1998 (KDT)
No Plan.

Login: yyc Name: yoonyc
Directory: /home1/yyc Shell: /bin/bash
Office: woosuk
Last login Wed Dec 30 09:34 (KDT) on tty7 from comstat1
Mail last read Wed Dec 16 21:38 1998 (KDT)
No Plan.
```

### 6.10 hostname

호스트에 부여된 이름을 출력한다.

```
cyber:~$hostname
cyber.woosuk.ac.kr
```

### 6.11 uname

현재 사용중인 운영체제와 커널의 컴파일 정보(-a) 등을 출력한다.

```
cyber:~$uname -a
Linux cyber.woosuk.ac.kr 2.4.20-8 #1 Thu Mar 13 17:54:28 EST 2003 i686 i686 i386 GNU/Linux
```

### 6.12 shutdown

```
root:~#reboot ←시스템을 재부팅한다.
root:~#shutdown -r now ←시스템을 재부팅한다.
root:~#shutdown -h now ←시스템을 완전히 종료한다.
```

### 6.13 w

로그인 되어있는 사람과 현재 실행중인 작업의 종류를 출력한다.

```
cyber:~$w
20:44:44 up 7:45, 5 users, load average: 0.07, 0.07, 0.02
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
root :0 - 1:01pm ? 0.00s 0.14s /bin/sh /usr/bi
root pts/1 - 1:01pm 7:43m 0.00s 0.44s kdeinit: kwrite
root pts/2 - 8:40pm 3:56 0.05s 0.05s /bin/bash
root pts/0 - 8:41pm 3:42 0.06s 0.06s -bash
yoon pts/3 css 5:44pm 0.00s 0.20s 0.01s w
```

'finger @호스트'는 현재 어떤 사람이 원격지 호스트를 사용하고 있는지를 나타낸다.

```
% finger @cyber
[cyber.woosuk.ac.kr]
Login Name Tty Idle Login Time Office Office Phone
root root *1 22d Nov 24 23:58
root root *p1 53d Dec 7 17:20
root root p0 47d Dec 7 17:20 (:0.0)
root root *p5 8d Dec 8 18:53
yoonyc Yoon YeoChang p9 1:20 Dec 30 09:40 (cyber)
```

### 7.4 ftp

인터넷 상에서 FTP 사이트에 접근하는 방법은 두 가지가 있다. 먼저 파일과 디렉토리의 공개적인 접근을 하게 해주는 anonymous FTP 가 있다. 이 경우에는 파일에 접근하기 위해 anonymous 로 로그인 하고 암호는 사용자 개인의 전자우편주소가 된다. 따라서 anonymous FTP 는 누구나 접근이 가능하다. 또 다른 경우는 정당한 계정과 암호를 부여 받은 특정한 사용자에게만 접근이 허용되는 FTP 가 있다. 이 경우에는 개인의 ID 와 비밀번호를 사용해야 한다.

만약에 anonymous FTP 사이트인 cyber.woosuk.ac.kr 에서 /pub/win95-appl/ftp/ws\_ftp95.exe 파일을 가져온다면 다음과 같은 절차가 필요하다.

## 7. 인터넷 명령어

### 7.1 ping

특정 호스트와의 연결상태를 확인할 수 있다.

```
cyber:~$ping www.woosuk.ac.kr
PING www.woosuk.ac.kr (210.93.6.3) 56(84) bytes of data:
64 bytes from www.woosuk.ac.kr (210.93.6.3): icmp_seq=1 ttl=252 time=3.59 ms
64 bytes from www.woosuk.ac.kr (210.93.6.3): icmp_seq=2 ttl=252 time=0.604 ms

--- www.woosuk.ac.kr ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9093ms
rtt min/avg/max/mdev = 0.550/0.894/3.599/0.904 ms
```

### 7.2 nslookup

인터넷 IP 번호를 대응하는 도메인 이름으로, 도메인 이름을 대응하는 IP 번호로 바꿔주는 기능을 한다.

```
% ftp cyber.woosuk.ac.kr ← ①
Connected to cyber.woosuk.ac.kr.
220 cyber.woosuk.ac.kr FTP server .
Name (cyber.woosuk.ac.kr:yoonyc): anonymous ← ②
331 Guest login ok, send your complete e-mail address as password.
Password: ← ②
230-Welcome to the Cyber's FTP Server.
230-Please send any questions,comments,or problem reports about this
server to yoonyc@cyber.woosuk.ac.kr
230-***** since June 10, 1996 *****
230 Guest login ok, access restrictions apply.
ftp> cd /pub/win95-appl/ftp ← ③
250 CWD command successful.
ftp> ls ← ④
200 PORT command successful.
150 Opening ASCII mode data connection for file list.
wsftp32a.zip
Cupertino.Beta2.zip
32cftp17.EXE
ws_ftp95.exe
226 Transfer complete.
63 bytes received in 0.027 seconds (2.3 Kbytes/s)
ftp> bin ← ⑤
200 Type set to I.
ftp> hash ← ⑥
Hash mark printing on (8192 bytes/hash mark).
ftp> get ws_ftp95.exe ← ⑦
200 PORT command successful.
150 Opening BINARY mode data connection for ws_ftp95.exe (875602
bytes).
#####
226 Transfer complete.
local: ws_ftp95.exe remote: ws_ftp95.exe
875602 bytes received in 3.1 seconds (2.7e+02 Kbytes/s)
ftp> quit ← ⑧
221 Goodbye.
%
```

① 연결하기

FTP 사이트에 연결하기 위해서는 먼저 이름을 알아야 한다. 일  
반적인 방법은 다음과 같다.

```
% ftp hostname.domainname
```

본 예제에서 hostname.domainname 은 cyber.woosuk.ac.kr 이다. 실  
제적인 사용방법은 연결될 시스템의 종류에 따라 다양하다. 잠시  
후 다음과 같은 메시지가 응답되면 성공적으로 연결이 된 것이다.

```
% ftp cyber.woosuk.ac.kr
```

```
Connected to cyber.woosuk.ac.kr.
```

② 로그인 하기

접속이 완료되면 사용자로 하여금 적절한 ID 와 비밀번호를 묻  
게 된다. 본 예제에서는 anonymous FTP 를 고려하고 있으므로 다  
음과 같이 입력한다.

```
Name (cyber.woosuk.ac.kr:yoonyc): anonymous
```

```
331 Guest login ok, send your complete e-mail address as password.
```

```
Password: ws981234@woosuk.woosuk.ac.kr
```

Password 항목에서 사용자의 전자우편주소를 입력한다. 실제 계

정을 사용하는 사용자의 비밀을 보장하기 위하여 전자우편주소가  
화면에 나타나지는 않는다. 입력이 끝나면 FTP 서버에서 보내온  
간단한 메시지를 받고 다음과 같이 접속을 하게 된다.

```
230 Guest login ok, access restrictions apply.
```

```
ftp>
```

③ 디렉토리 바꾸기(cd)

FTP 연결 초기에 사용자의 디렉토리는 'top-level' 디렉토리일 것  
이다. 대부분의 파일들은 하부 디렉토리에 위치하며 통상 /pub 에  
있게 된다. 현재 디렉토리를 변경하기 위하여 cd 명령을 이용한다.  
'pub' 디렉토리로 변경하기 위해서는 다음과 같이 입력한다.

```
ftp> cd /pub
```

그러면 다음과 같은 메시지가 나타나며 디렉토리가 변경된다.

```
250 CWD command successful.
```

```
ftp>
```

이 예제에서 다시 /pub/win95-appl/ftp 디렉토리로 변경은 다음과  
같다.

```
ftp> cd win95-appl/ftp
```

```
250 CWD command successful.
```

만약에 여기서 한단계 위의 디렉토리로 올라가려면 cd ..명령을  
한다

```
ftp> cd ..
```

그리고 디렉토리를 많이 변경하였을 때에는 현재의 작업 디렉토리  
를 확인해 볼 필요가 있는데 명령은 pwd 이다.

```
ftp> pwd
```

```
257 "/pub/win95-appl" is current directory.
```

④ 파일목록 보기(dir)

ftp> 프롬프트에서 사용자는 다양한 명령어를 입력할 수 있다.  
'dir'은 현재 디렉토리에 있는 파일을 보여준다.

```
ftp> dir
```

```
200 PORT command successful.
```

```
150 Opening ASCII mode data connection for /bin/ls.
```

```
total 2129
```

```
drwxr-xr-x 2 root root 1024 Jul 10 1997 .
```

```
drwxr-xr-x 14 root root 1024 Jun 2 11:48 ..
```

```
-rwxr--r-- 1 root root 494802 Nov 6 1996 32cftp17.EXE
```

```
-rwxr--r-- 1 root root 875602 Jul 10 1997 ws_ftp95.exe
```

```
-rwxr--r-- 1 root root 480053 Oct 1 1996 wsftp32a.zip
```

226 Transfer complete.

412 bytes received in 0.022 seconds (19 Kbytes/s)

ftp>

'dir'은 'ls -la'와 같이 파일의 속성, 크기등을 함께 나타내고 'ls'는 파일 이름만 간단하게 출력한다.

⑤ 전송 모드 설정(ascii, binary)

전송할 파일이 텍스트 모드(\*.txt, .dat 등)로 저장된 파일이라면 전송 모드를 ascii 로 설정해야 하고, binary 파일(\*.exe, \*.hwp, \*.doc, \*.zip 등)이라면 전송 모드를 binary 로 설정해야 한다. 각 명령은 한번만 수행하면 된다. 즉 사용자가 binary 를 입력하면 그 연결상태에 있는 모든 전달은 ascii 가 입력되기 전까지 binary 모드에서 수행된다. 두 명령의 사용 방법은 다음과 같다.

ftp> ascii ← 간단히 asc 로 입력해도 된다.

200 Type set to A.

ftp> binary ← 간단히 bin 으로 입력해도 된다.

200 Type set to I.

만약에 binary 파일을 텍스트 모드로 전송 받으면 데이터가 손실 변형된다. 그러나 텍스트 파일인 경우에는 텍스트 모드와 binary 모드의 어느 한 경우로 설정해도 데이터의 변형 없이 전송 받을 수 있다.

⑥ 전송과정의 확인(hash)

사용자가 받을 파일의 크기가 매우 큰 경우에 전송과정을 표시 (#)할 수 있으면 종료 시점을 예측할 수 있게 되어 편리할 수 있다. 만약에 그 확인을 하지 않으면 전송 중에 마치 시스템이 정지된 상태를 계속 유지하기 때문에 답답할 수도 있다. 전송과정의 확인 절차는 다음 명령으로 수행한다. 예제에서 한 개의 #표시는 8192 바이트를 나타내고 있다.

ftp> hash

Hash mark printing on (8192 bytes/hash mark).

ftp> get ws\_ftp95.exe

200 PORT command successful.

150 Opening BINARY mode data connection for ws\_ftp95.exe (875602 bytes).

#####

226 Transfer complete.

local: ws\_ftp95.exe remote: ws\_ftp95.exe

875602 bytes received in 3.1 seconds (2.7e+02 Kbytes/s)

⑦ 파일 가져오기(get)와 파일 올리기(put)

실제적인 파일 전달이 이루어지기 위해서는 get, put 명령이 필요하다. 원격지의 컴퓨터로부터 사용자의 시스템으로 파일을 가져오기 위해서는 다음과 같은 형식을 따른다.

ftp> get filename

여기서 filename 에는 원격지 시스템의 파일이름을 적는다. 따라서 예제에서 ws\_ftp95.exe 를 가져온다면 디렉토리를 변경 후에 다음과 같이 명령한다.

ftp> get ws\_ftp95.exe

파일 전송의 또 다른 방법으로 put 명령은 사용자의 시스템으로부터 원격지의 시스템으로 파일을 전달해 준다. 단 원격지의 디렉토리에 파일을 쓸 수 있는 권한이 주어진 경우에만 해당되며, 명령은 다음과 같은 형식을 따른다.

ftp> put filename

mget 과 mput 명령어는 wildcard(\*)를 사용하여 한번에 다수 개의 파일을 전송할 수 있도록 한다. 예를 들어 f 로 시작되는 모든 파일을 가져오기 위해서는 다음과 같이 명령한다.

ftp> mget f\*

또한 모든 파일을 가져오기 위해서는 다음과 같이 명령한다.

ftp> mget \*

같은 방법으로 사용자 파일 중 확장자가 .c 인 모든 파일을 전송하기 위해서는 다음과 같이 명령한다.

ftp> mput \*.c

여러 개의 파일을 전송받는 mget 의 사용 예는 다음과 같다. 예를 들어 확장자가 exe 인 모든 파일을 받고 싶다면 다음과 같이 입력한다.

```
ftp> mget *.exe
mget ws_ftp95.exe? y
mget cp32e45.exe? y
```

그러면 파일 하나씩 확인을 하고 전송을 한다. 여기서 ws\_ftp95.exe 파일을 받고 싶으면 y 를 입력한다. 이와 같은 확인 절차 없이 모든 파일을 전송 받고 싶으면 prompt 를 입력한다.

```
ftp> prompt
Interactive mode off.
```

또한 다시 prompt 를 입력하면 확인절차가 다시 나타나게 된다.

```
ftp> prompt
Interactive mode on.
```

⑧ 끝내기(exit)

파일 전송을 끝내고 싶을 때는 quit 또는 bye 를 입력한다.

```
ftp> quit
221 Goodbye.
%
```

7.5 mail

메일 도구로는 여러 가지가 있지만 간단하게 쓰기 위한 기본 메일 도구로는 mail 이 있다. 다음의 예는 core.woosuk.ac.kr 에 있는 yoonyc 라는 사용자에게 메일을 쓰는 예이다. 이 메일의 제목은 'mail test'이고, 본문의 내용은 'this is mail test'이다. 본문의 내용을 다 입력한 후에 [control] + d 를 누르면 mail 이 종료되고 곧바로 보내진다. [control] + d 와 같은 명령으로써 .을 새로운 줄의 첫번째 열에 입력하면 된다.

```
% mail yoonyc@woosuk.ac.kr
Subject : mail test
this is mail test.
[control]과 d 키를 동시에 누른다
%
```

메일을 보내는 과정을 확인하고 싶으면 다음과 같이 명령한다.

```
% mail -v yoonyc@core.woosuk.ac.kr
Subject: test2
this is mail test2
.
EOT
yoonyc@core.woosuk.ac.kr... Connecting to core.woosuk.ac.kr. (smtp)...
220 core.woosuk.ac.kr ESMTP Sendmail 8.8.8H1/8.8.8; Tue, 7 Jul 1998
17:07:13 +0900 (KST)
>>> HELO comstat1.woosuk.ac.kr
250 core.woosuk.ac.kr Hello root@comstat1.woosuk.ac.kr
[203.237.211.166], pleased to meet you
>>> MAIL From:<root@comstat1.woosuk.ac.kr>
250 <root@comstat1.woosuk.ac.kr>... Sender ok
>>> RCPT To:<yoonyc@core.woosuk.ac.kr>
250 <yoonyc@core.woosuk.ac.kr>... Recipient ok
>>> DATA
354 Enter mail, end with "." on a line by itself
>>> .
250 RAA12772 Message accepted for delivery
yoonyc@core.woosuk.ac.kr... Sent (RAA12772 Message accepted for
delivery)
Closing connection to core.woosuk.ac.kr.
>>> QUIT
221 core.woosuk.ac.kr closing connection
%
```

받은 메일을 읽고자 할 때에는 mail 을 실행한다.

```
% mail
Mail version 5.5 6/1/90. Type ? for help.
"/var/spool/mail/root": 2 message 2 new
>N 1 root@comstat1.woosuk Tue Jul 7 16:34 15/841 "test"
>N 2 root@comstat1.woosuk Tue Jul 7 16:38 24/918 "test1"
& 1 ← 1 번 mail 을 읽는다.
Message 1:
From root@comstat1.woosuk.ac.kr Tue Jul 7 16:34:54 1998
Date: Tue, 7 Jul 1998 16:34:54 +0900
From: root <root@comstat1.woosuk.ac.kr>
To: yoonyc@core.woosuk.ac.kr
Subject: mail test

This is mail test.

&
```

간단한 도움말을 보고 싶을 때는 ?를 입력한다.

```
& ?
Mail Commands
t <message list> type messages
n goto and type next message
e <message list> edit messages
f <message list> give head lines of messages
d <message list> delete messages
s <message list> file append messages to file
u <message list> undelete messages
R <message list> reply to message senders
r <message list> reply to message senders and all recipients
pre <message list> make messages go back to /usr/spool/mail
m <user list> mail to specific users
q quit, saving unresolved messages in mbox
x quit, do not remove system mailbox
h print out active message headers
! shell escape
cd [directory] chdir to directory or home if none given
A <message list> consists of integers, ranges of same, or user names
separated by spaces. If omitted, Mail uses the last message typed.
A <user list> consists of user names or aliases separated by spaces.
Aliases are defined in .mailrc in your home directory.
&
```

위 명령은 시스템마다 약간의 차이가 있을 수 있다.

7.6 uuencode, uudecode

메일을 보낼 때 binary 파일을 같이 보내야 할 경우가 있다. 이 때 바이너리파일을 텍스트파일로 변환하는 명령이 uuencode 이다. 형식은 uuencode [변환할파일] [변환후이름]이다. yoon.gif 파일을 보낼 경우에는 다음과 같다.

```
% uuencode yoon.gif yoon.gif > yoon.uu
```

yoon.gif 파일을 yoon.gif 라는 이름으로 변환한 후에 yoon.uu 파일에 저장하라는 뜻이다. 따라서 mail 을 보낼 때 위 텍스트 파일을 본문에 삽입하여 보내면 된다. 메일을 받은 쪽에서는 해당 메일을 저장한 후에 다음과 같이 하면 원래의 파일로 다시 변환된다. 만약에 저장된 메일이름이 remail 이라고 가정하면 다음과 같다.

```
% uudecode remail
% ls
% yoon.gif
```

메일로 보내는 작업을 한 번에 하려면 다음과 같이 입력하면 된다.

```
%uuencode yoon.gif yoon.gif | mail yoonyc@core.woosuk.ac.kr
```

미리 작성해 둔 텍스트 파일 test.txt 를 전자우편으로 보내려면 다음과 같다.

```
% mail yoonyc@core.woosuk.ac.kr < test.txt
```

### 8. 통신 명령어

#### 8.1 write

간단한 통신 명령어이다. 만약 askim 이 현재 접속중인 yoon 이 라는 사람과 직접 통신을 원한다면 다음과 같다.

(예) [write] [yoon]

```
$ write yoon
```

그러면 askim 의 화면에 다음과 같이 나타난다.

```
Yoon patt (pts/6) [ 12 19 17:55:52 ] ...
```

이때 yoon 은 다음 명령으로 통신을 수락하고 서로 입력한 대화는 상대방의 화면에 출력된다.

```
$ write askim
```

대화를 끝내려면 O-O(모두 끝이라는 뜻 Over and Over)를 타이프 한 후, ^d 를 누른다. 여기서 ^는 [Ctrl]키를 의미한다. 참고로 @는 [Alt]키를 말하며 ~는 [Shift]키를 의미한다.

#### 8.2 talk

write 보다 좀 더 진보된 통신 방법으로는 talk 가 있다. 만약 yoon 이 현재 접속중인 askim 이라는 사람과 직접 통신을 원한다면 다음과 같다.

```
$ talk askim
```

그러면 askim 의 화면에 다음과 같이 나타난다.

```
Message from Talk_Daemon@comstat1 at 18:05 ...
Talk: connection requested by yoon@comstat1.
Talk: respond with: talk yoon@comstat1.
```

이때 askim 은 다음 명령으로 통신을 수락하고 서로 입력한 대화는 화면이 양분되면서 위칸은 자신이 입력한 내용이 아래칸은 상대방이 보내온 내용이 출력된다.

```
$ talk yoon
```

통신을 끝내려면 ^d 를 누른다.

#### 8.3 mesg

어떤 작업 중에 다른 사람으로 부터 통신 접속 메시지가 날라 오면 곤란할 경우가 있다. 이때 mesg 는 통신 접속을 차단하는 명령이다.

```
$ mesg n
```

통신 접속을 다시 허용하려면 다음과 같다.

```
$ mesg y
```

#### 8.4 wall

시스템에 접속중인 모든 사람에게 동일한 메시지를 보내고 싶을 때는 다음과 같다.

```
$ wall
```

### 9. 셸

셸이란 시스템과 사용자의 중간에 위치하여 사용자가 입력하는 명령어를 시스템이 실행할 수 있는 기계어로 번역하는 사용자 인터페이스이다.

#### 9.1 와일드 카드

명령어를 입력할 때에 와일드 카드를 이용하여 해당 파일을 총칭할 수 있다. 즉 \*는 모든 것을 뜻하고, ?는 어떤 한 문자를 뜻한다. 그리고 [...]는 []안의 어떤 문자와도 결합될 수 있음을 뜻한다.

(예 1)

```
x*y ← xabcdfy, xay, xabcd1234y, .....
x?y ← xay, xby, x1y, x2y.....
x[ab]y ← xay, xby.
```

(예 2)

```
rm * ← 현재의 디렉토리에서 모든 파일을 삭제한다.
cp -r * /home ← 현재의 디렉토리에서 모든 파일과 서브디렉토리의 내용을 /home 으로 복사한다.
```

#### 9.2 표준입력과 출력

실행의 입력과 출력 결과등에 대한 명령으로는 입력(<), 출력(>),



추가(>>), 표준에러(2>), 결과값 전달() 등이 있다.

(예)

test << in > out ← in 파일의 내용을 test 에 추가하여 out 파일을 생성한다.

ls -al | more ← ls -al 의 결과를 한 화면씩 보여준다.

test 2> error ← test 파일을 실행하며 나오는 에러는 error 에 기록한다.

### 9.3 작업조정

유닉스 시스템에서는 여러 가지의 작업을 동시에 수행할 수 있다. 이때 작업자와 직접 대화식으로 프로그램이 동작하는 배경을 foreground 라고 하고, 작업자와 독립적으로 동작하는 프로그램의 배경을 background 라고 한다.

- bg (background job)

새롭게 입력하여 실행되는 명령의 이면에서 동작하는 프로세스를 의미한다. 일반적으로 명령이 실행중일 때에는 프롬프트가 반환되지 않기 때문에 새로운 명령을 실행 할 수 없다. 그러나 명령 라인의 맨 뒤에 &를 붙이면, 명령의 종료를 기다리지 않고 프롬프트가 반환되어 새로운 명령을 실행 할 수 있게 된다. 이와 같이 셸과 병행하여 이면에서 동작하는 프로세스를 background job 이라고 한다.

- fg (foreground job)

단말기와 대화적으로 처리를 하는 것을 의미한다.

어떠한 작업을 bg 로 실행하고 싶을 경우는 끝에 &를 붙인다. 예를 들어 xyz 와 같은 실행 프로그램을 bg 로 실행시키고자 할 경우는 다음과 같다.

```
$ xyz &
[2] 500
```

여기서 xyz 를 fg 로 돌리고 싶을 경우는 다음과 같다.

```
$ fg %2
xyz
```

다시 bg 로 돌리고 싶다면 다음과 같다.

```
$ fg %2
xyz
← '[control] + Z 키를 누른다'
[2]+ Stopped (user) xyz
$ bg
[2]+ xyz &
```

xyz 를 강제로 종료 시키려면 다음과 같다. 여기서 -9 는 강제로 해당 작업을 종료하라는 의미이다.

```
$ kill -9 %2
[2]- Killed xyz
```

## 10. 시스템 관리 명령어

### 10.1 파일 형식

#### 10.1.1 유닉스 파일 구조

유닉스는 여러 사람이 동시에 시스템을 사용할 수 있는 다중 사용자 환경이다. 그렇기 때문에 하나의 파일마다 사용자(소유자), 그 사용자가 속한 그룹, 그 외의 기타 사용자에게 각각의 파일 접근 권한을 분리하여 부여할 필요가 있다. 그렇지 않다면 힘들여 작성한 자신의 파일을 다른 사람이 내용을 변경하거나 삭제할 수도 있게 된다. 혹은 관련이 있는 집단의 사람에게만 권한을 줄 수 있도록 그룹으로도 묶어 놓을 수 있다. 다음과 같은 파일 구조를 이용하여 권한 조정에 관한 내용을 살펴보자.

```
$ ls -la
drwxr-xr-- 3 yoonyc users 1024 Jul 18 1997 home-pic/
-rw----- 1 yoonyc users 27589 Jun 12 15:50 pager.exe
-rw-r--r-- 1 yoonyc users 0 Feb 2 11:11 php_fi.txt
lrwxrwxr-x 1 root root 8 Aug 6 06:32 yuvtoppm->ppmmerge
drwxrwxr-x 2 yoonyc users 1024 Jul 15 1997 tcl/
```

각 행에서 첫번째 열은 파일의 종류를 의미한다. 즉 -는 일반파일, d 는 디렉토리 그리고 l 은 링크된 파일을 의미한다. 두번째 이후의 열은 파일 사용의 허가권을 의미한다. 즉 r 은 읽을 수 있고, w 는 쓸 수 있고, x 는 실행할 수 있음을 의미한다.

#### 10.1.2 허가 (permissions)

다음과 같이 9 개의 필드로 나누어서 디렉토리 home-pic 을 살펴보자.

파일 종류	사용자 허가	그룹 허가	기타 허가	사용자 이름	그룹 이름
d	rwx	r-x	r--	yoonyc	users

위 예에서 [home-pic]는 디렉토리이며 사용자는 [yoonyc]이며 그룹은 [users]에 속한다. 그리고 사용자 [yoonyc]는 읽기(r) 쓰기(w) 실행(x)할 수 있으며 그룹 [users]에 속한 사람들(또는 프로그램들)은 읽기(r)와 실행(x)만 가능하며 기타 사용자는 읽기(r)만 가능하다.

다.

### 10.1.3 허가 변경 명령(chmod)

기존의 허가권을 변경하기 위한 명령이다. 허가권의 변경은 자신의 파일에 한해서만 가능하다.

허가권의 변경을 위한 일반적인 명령방법은 다음과 같다.

[chmod] [허가코드] [대상파일]

허가코드	사용자 허가			그룹 허가			기타 허가		
	읽기+쓰기+실행			읽기+쓰기+실행			읽기+쓰기+실행		
문자	r	w	x	r	w	x	r	w	x
421 코드	4	2	1	4	2	1	4	2	1

허가 코드는 읽기→ 4, 쓰기 → 2, 실행 → 1 이다. 따라서 읽기+쓰기+실행=7(rwx), 읽기+쓰기=6(rw-), 읽기+실행=5(r-x), 읽기=4(r--), 쓰기=2(-w-), 실행=1(--x) 그리고 허가없음=0(---)이다. 이 값들을 사용자 허가, 그룹 허가, 기타 사용자 허가를 위하여 각각 부여하면 된다.

(예) home-pic 라는 디렉토리를 모든 사용자에게 읽기 권한만 주고 싶을 경우에는 다음과 같다.

```
$ chmod 444 home-pic
$ ls -la
dr--r--r-- 2 yoonyc users 1024 Jul 18 1997 home-pic/
```

(예) home-pic 라는 디렉토리를 모든 사용자에게 읽기와 실행 권한만 주고 싶을 경우에는 다음과 같다.

```
$ chmod 555 home-pic
$ ls -la
dr-xr-xr-x 2 yoonyc users 1024 Jul 18 1997 home-pic/
```

(예) home-pic 라는 디렉토리를 사용자에게 모든 권한을, 그룹에게는 읽기와 실행 권한을, 제 3 자에게는 읽기권한만 주고 싶을 경우에는 다음과 같다.

```
$ chmod 754 home-pic
$ ls -la
drwxr-xr-- 2 yoonyc users 1024 Jul 18 1997 home-pic/
```

(예) home-pic 라는 디렉토리를 시스템관리자(superuser)에게만 모든 권한을 주고, 사용자 허가, 그룹 허가, 기타 허가는 모두 주지 않을 경우는 다음과 같다.

```
$ chmod 000 home-pic
$ ls -la
d----- 2 yoonyc users 1024 Jul 18 1997 home-pic/
```

### 10.1.4 소유자와 그룹 변경(chown, chgrp)

파일의 소유자와 그룹을 변경할 수 있으며 주로 시스템관리자에 의해 수행된다. 다음의 예는 소유자와 그룹이 모두 root 이다. 여기서 yoon.BMP 를 yoonyc 소유로 바꾸는 명령은 다음과 같다.

```
$ ls -la
total 1034
drwxr-xr-x 3 yoonyc root 1024 Jul 2 1997 ./
drwxr-xr-x 22 yoonyc users 2048 Jun 12 15:50 ../
-rw-r--r-- 1 root root 32280 Jun 9 1997 yoon.BMP
$ chown yoonyc yoon.BMP
```

소유자가 바뀐 것을 확인한 후, 다시 그룹을 users 로 바꾸는 명령은 다음과 같다.

```
$ ls -la
total 1034
drwxr-xr-x 3 yoonyc root 1024 Jul 2 1997 ./
drwxr-xr-x 22 yoonyc users 2048 Jun 12 15:50 ../
-rw-r--r-- 1 root root 32280 Jun 9 1997 libc.tar.gz
-rw-r--r-- 1 yoonyc root 32280 Jun 9 1997 yoon.BMP
$ chgrp users yoon.BMP
```

다음 예를 통하여 그룹이 바뀐 것을 알 수 있다. 이제 소유자와 그룹을 각각 바꾸지 않고 한꺼번에 허가를 변경하는 방법은 다음과 같다.

```
$ ls -la
total 1034
drwxr-xr-x 3 yoonyc root 1024 Jul 2 1997 ./
drwxr-xr-x 22 yoonyc users 2048 Jun 12 15:50 ../
-rw-r--r-- 1 root root 32280 Jun 9 1997 test.tar.gz
-rw-r--r-- 1 yoonyc users 32280 Jun 9 1997 yoon.BMP
$ chown yoonyc.users test.tar.gz
$ ls -la
total 1034
drwxr-xr-x 3 yoonyc root 1024 Jul 2 1997 ./
drwxr-xr-x 22 yoonyc users 2048 Jun 12 15:50 ../
-rw-r--r-- 1 yoonyc users 32280 Jun 9 1997 test.tar.gz
-rw-r--r-- 1 yoonyc users 42280 Jun 9 1997 yoon.BMP
```

현재 디렉토리의 모든 파일들을 소유자는 yoonyc, 그룹은 users 로 일괄적으로 바꾸는 예는 다음과 같다.

```
$ chown yoonyc.users *
```

## 10.2 프린터 사용

### 10.2.1 lp

프린터에 파일을 출력하는 명령이다. 다음의 예는 /etc/passwd 파일을 출력한다.

```
% lp /etc/passwd
```

### 10.2.2 lpq

프린터의 현재 작업 상황을 출력한다. 다음의 예에서 ps 는 현재 시스템의 기본 스플러 이름이다.

```
% lpq
ps is ready and printing
Rank  Owner   Job    Files      Total Size
Active yoonyc   72    /etc/passwd 636 bytes
```

### 10.2.3 lprm

출력을 취소한다. 이때는 lprm 'job-number'와 같은 형식을 따른다.

```
% lpq
ps is ready and printing
Rank  Owner   Job    Files      Total Size
Active yoonyc   72    /etc/passwd 636 bytes
% lprm 73
```

## 10.3 시스템 운영

### 10.3.1 운영 도구

각 회사별로 운영자를 위한 시스템 운영 도구가 준비되어 있다. 운영 도구에는 SUN 의 admintool, HP 의 sam 그리고 IBM 의 smit 등이 있다. 이와 같은 도구를 사용하여 시스템을 관리하면 편리하다. 그러나 운영도구가 없을 경우에는 직접 명령어를 이용하여 관리하면 된다.

### 10.3.2 사용자 추가

시스템에 사용자를 추가하는 방법은 다음과 같다. 먼저 askim 이라는 사용자를 추가하는 경우를 살펴보자.

- 1) /etc/group 에 사용자를 추가 한다.

```
noaccess::60002:
nogroup::65534:
users::200:aslee,askim
anonymous::101:
```

- 2) /etc/passwd 에 사용자를 추가 한다.

(예) 사용자 ID::UID:GID:Home-directory:shell

```
nobody4:x:65534:65534:SunOS 4.x Nobody:/:
askim::203:200::/home/askim:/bin/bash
bspaek:x:201:200::/home/bspark:/bin/bash
aslee:x:202:200::/home/aslee:/sbin/bash
ftp:x:1001:101::/home/ftp/home:/nosuchshell
```

- 3) 사용자의 홈 디렉토리를 만든다.

```
% mkdir /home/askim
```

- 4) 사용자의 홈 디렉토리를 사용자 소유로 한다.

```
% chown askim /home/askim
```

- 5) 사용자의 홈 디렉토리를 소속 그룹으로 소유시킨다.

```
% chgrp users /home/askim
```

### 10.3.3 mount

마운트는 각각의 파일 시스템 (하드디스크, CD-ROM, 플로피디스크 등)을 디렉토리 구조에 연결하여 포함시키는 작업이다. 하드디스크의 경우 시스템이 부팅될 때 자동으로 /etc/mnttab 혹은 /etc/fstab 등을 참조하여 마운트 되며 사용자는 CD-ROM 이나 플로피디스크등과 같은 기구의 연결에 주로 사용 한다. 시스템과 운영체제에 따라 옵션이나 사용법이 다른점이 있으나 일반적인 명령구조는 다음과 같다.

```
mount [file-system-type] [device] [mount-dir]
```

- 1) mount

```
$ mount -t msdos /dev/fd0 /mnt/floppy
$ mount -t iso9660 /dev/cdrom /mnt/cdrom
```

- 2) umount

디렉토리와 연결되었던 시디롬이나 플로피 디스크를 분리시켜 주는것을 말한다. 플로피나 시디롬을 다 쓰고 나면 항상 언마운트를 시켜주어야 한다. 일반 형식은 다음과 같다.

```
umount mount-dir
```

```
$ umount /mnt/cdrom
$ umount /mnt/floppy
```

### 10.3.4 테이프 드라이브의 사용

하드디스크의 자료를 정기적으로 백업할 경우 주로 사용 한다. 테이프 드라이브의 종류로는 테이프의 직경에 따라 4mm(DAT 라고도 하며 보통 2GB 이상), 8mm(일반 카세트 테이프 정도의 크기, 보통 2GB 이상), 16mm(비디오 테이프 정도의 크기, 150MB)가 있다.

- 1) 테이프 제어명령(mt)의 일반 형식은 다음과 같다.

```
mt [ -f tapename ] command... [ count ]
```

Solarix2.5 에서 /home 을 테일에 백업할 경우는 다음과 같다.

```
mt -f /dev/rmt/0m rewind ← 테이프를 되감는다.
```

```
tar cvf /dev/rmt/0m /home ← /home 를 테이프에 저장한다.
```

```
tar xvf /dev/rmt/0m ← 테이프의 내용을 현 디렉토리에 푼다.
```

```
mt -f /dev/rmt/0m offline ← 테이프를 꺼낸다.
```

### 10.3.5 파일 시스템 점검(fsck)

파일 시스템을 점검하고 손상 부위를 보수 한다. 다음의 예는 손상부위를 사용자에게 묻지않고 전부 보수한다.

```
% fsck -Y
** /dev/rdisk/c0t0d0s5
** Currently Mounted on /opt
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
** Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
1908 files, 164286 used, 123609 free (441 frags, 15396 blocks, 0.1%
fragmentation)FILE SYSTEM STATE IN SUPERBLOCK IS WRONG;
FIX? yes** Phase 5 - Check Cyl groupsFILE SYSTEM STATE IN
SUPERBLOCK IS WRONG; FIX? yes** Phase 5 - Check Cyl groups
```

### 10.4 Network 명령

#### 10.4.1 ifconfig

ifconfig 는 시스템 부팅 시에 시스템에 장착된 각각의 인터페이스의 네트워크 주소를 정의하며 인터페이스의 주소를 재정의 할 때 사용한다. 다음과 같이 옵션 없이 사용하면 현재의 네트워크 인터페이스에 구성을 보여 준다.

```
$ ifconfig eth0 203.237.211.160 netmask 255.255.255.0
```

현재의 interface 를 취소하고 새로 정의할 경우는 다음과 같다.

```
$ ifconfig elx0 down
$ ifconfig elx0 "ip-address" up
```

#### 10.4.2 netstat

현재의 network 상태를 출력한다. 옵션 r 은 라우팅 테이블을 보여 준다. 옵션 n 은 도메인 네임의 형태를 ip 어드레스 형태로 보여 준다.

```
$ netstat -nr
Kernel routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
203.237.211.0 0.0.0.0 255.255.255.0 U 0 0 1956 eth0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 18 lo
0.0.0.0 203.237.211.1 0.0.0.0 UG 1 0 942 eth0
```

#### 10.4.3 route

라우팅 테이블을 직접 조작한다. 기본 게이트웨이를 지정하는 방법은 다음과 같다.

```
$ route add default "ip-address" 1
```

#### 10.4.3 traceroute

호스트와 호스트 사이의 라우팅 과정을 추적한다. 예에서는 사용자 호스트인 cyber.woosuk.ac.kr 에서 부터 원격지 호스트 yurim.skku.ac.kr 까지의 라우팅을 추적하고 있다.

```
$ traceroute yurim.skku.ac.kr
traceroute to yurim.skku.ac.kr (134.75.162.2), 30 hops max, 40 byte
packets
```

```
1 203.237.211.1 (203.237.211.1) 2.781 ms 2.838 ms 2.913 ms
2 core (210.93.6.2) 0.775 ms 1.261 ms 0.791 ms
3 147.47.79.1 (147.47.79.1) 4.651 ms 4.612 ms 4.69 ms
4 chonbuk-gw2.chonbuk.ac.kr (192.203.139.200) 105.654 ms
18.966 ms 20.408 ms
5 134.75.183.1 (134.75.183.1) 15.948 ms 13.059 ms 14.429 ms
6 baram.kreonet.re.kr (134.75.20.1) 14.963 ms 14.248 ms
13.778 ms
7 Seoul-Taejon-T3.kreonet.re.kr (134.75.3.2) 14.329 ms 15.871
ms 15.241 ms
8 skku-gurum.kreonet.re.kr (134.75.62.2) 23.574 ms 21.012 ms
19.346 ms
9 yurim.skku.ac.kr (134.75.162.2) 22.324 ms * 19.631 ms
```

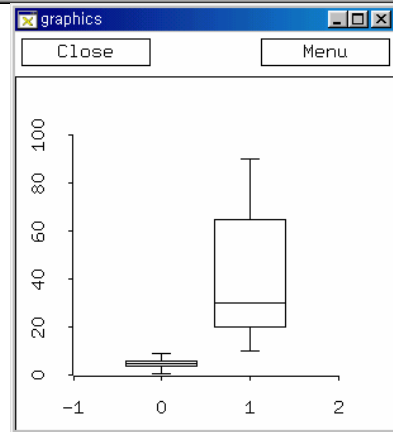
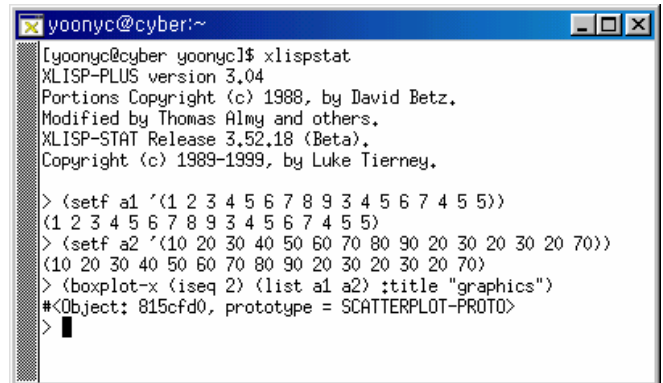
## 11. 프로그램 설치 및 활용

### 11.1 Xlisp-stat 설치해 보기

파일 다운로드: <http://www.xlispstat.org/dist/xlispstat-3-52-18.tar.gz>

```
현재 디렉토리 /usr/local/src 에 파일을 받음.
[yoonyc]$gzip - cd xlispstat-3-52-18.tar.gz | tar xvf -
[yoonyc]$cd xlispstat-3-52-18
[yoonyc]$./configure - prefix=/usr/local
[yoonyc]$make
[yoonyc]$make install
[yoonyc]$xlispstat
XLISP-PLUS version 3.04
Portions Copyright (c) 1988, by David Betz.
Modified by Thomas Almy and others.
XLISP-STAT Release 3.52.18 (Beta).
Copyright (c) 1989-1999, by Luke Tierney.
>
```

윈도우 98 에서 Xmanager 를 이용한 실행결과:



### 11.2 부팅시 출력되는 메시지 보기

간단히 Shift+PgUp/Dn 키를 눌러 지나간 화면을 볼 수 있다.

부팅시 출력되는 메시지는 /var/log/dmesg 파일에 매번 기록이 된다.

### 11.3 시스템에 대한 정보

/proc 디렉토리 아래 파일을 출력하면 간단한 정보를 알 수 있다.

```

/proc/cpuinfo      : cpu 정보
/proc/pci          : pci 정보
/proc/interrupts  : 사용중인 irq 정보
/proc/dma          : 사용중인 dma 채널
/proc/ioports     : 사용중인 i/o 포트 정보
/proc/filesystems : 현재 사용가능한 화일 시스템
/proc/meminfo     : 현재 사용중인 메모리
/proc/partitions  : 현재 사용중인 파티션
/proc/swaps       : 현재 사용중인 스왑 파티션 정보
/proc/version     : 현재 사용중인 커널 버전
/proc/devices     : 현재 사용중인 드라이버
/proc/cmdline     : 현재 사용중인 부트 이미지
    
```

```

root@cyber:/proc
root:/proc#more cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 15
model         : 1
model name    : Intel(R) Pentium(R) 4 CPU 1.70GHz
stepping     : 2
cpu MHz      : 1634.545
cache size   : 256 KB
fdiv_bug     : no
hlt_bug      : no
f00f_bug     : no
coma_bug     : no
fpu          : yes
fpu_exception: yes
cpuid level  : 2
wp           : yes
flags        : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm
bogomips    : 3361.65
root:/proc#
    
```

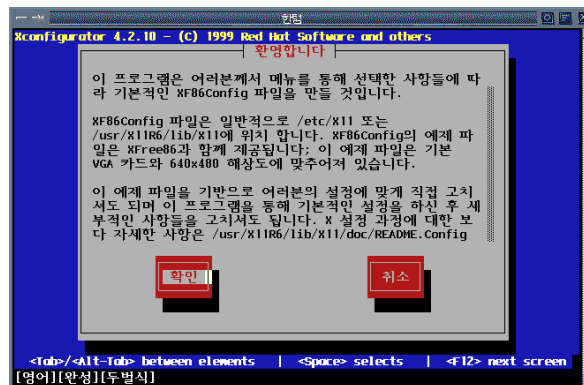
### 11.4 root 의 비밀번호를 잊어버린 경우

리눅스 싱글 모드로 부팅하신후 프롬프트 상태에서 passwd 를 실행한 후, 새로운 패스워드를 입력한다.

### 11.5 X 윈도우를 다시 설정

XFree86 의 설정을 변경 한다는 것은 결국 /etc/X11/XF86Config 파일을 편집하는 것이다. 편집을 위한 파일은 다음과 같다.

레드햇에서 제공하는 Xconfigurator 와 XFree86 에 포함된 F86setup 그리고 텍스트 기반의 XF86config 유틸리티.



그러나 RadHat 9.0 에서는 시스템 메뉴에서 통합 관리.

### 11.6 리눅스 부팅디스크 만들기

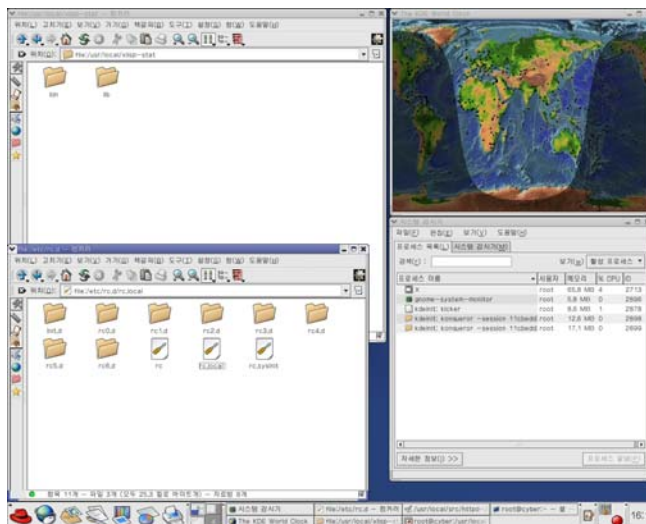
#fdformat /dev/fd0 : 플로피 디스크를 로우레벨 포맷.

#mkfs -t ext2 /dev/fd0 : ext2 파일 시스템으로 포맷.

### 11.7 화면을 캡처

RadHat 9.0 의 screen capture program 을 이용한다.

xv 를 이용하려면 <http://freshmeat.net> 에서 다운 설치할 수 있다.



### 11.8 X 윈도우에서 휠마우스를 사용

RadHat 9.0 에서는 기본적으로 이용 가능하다.

```

root@cyber:/etc/X11
Section "InputDevice"
Identifier "Mouse0"
Driver "mouse"
Option "Protocol" "IMPS/2"
Option "Device" "/dev/psaux"
Option "ZAxisMapping" "4 5"
Option "Emulate3Buttons" "yes"
EndSection
More--(67%)
    
```

와우 리눅스 6.2 에서는 휠마우스를 사용하기 위하여, imwheel 패키지가 포함되어 있다. imwheel 패키지가 설치 되어 있는지 확인한 후, /etc/X11/XF86Config 파일의 Pointer section 부분을 아래와 같이

변경한다.

Section "Pointer" Protocol "imps/2" ← PS2 방식의 휠마우스

Device "/dev/mouse" ← 마우스 장치명

ZAxisMapping 4 5 ← 휠마우스를 위한 추가 라인

### 11.9 root 계정으로 telnet 과 ftp 에 접속하기

보안상 root 계정의 접속을 허용하지 않는다.

일반 유저로서 접속한 후 su 명령을 통해 root 를 사용.

### 11.10 실행파일이 실행이 안되는 경우

파일의 이름앞에 ./ 를 붙여준다.

직접 실행시키는 방법은 다음과 같다.

자신의 홈 디렉토리의 .bash\_profile 에 정의되어 있는 PATH 에 ./

를 추가한다. 즉

```
PATH=$PATH:$HOME/bin:./
```

셸을 반영시키려면,

```
$.bash_profile 또는 logout → login 한다.
```

### 11.11 ls 시 컬러로 출력되는 폰트를 흑백으로 바꾸기

리눅스에서는 파일 유형에 따라 컬러를 구별한다.

~/bash\_profile 수정방법:

```
alias='ls --color=auto' ← 컬러
```

```
alias='ls' ← 흑백
```

### 11.12 사용자 계정의 백업및 복구

백업할 파일: /etc/passwd, /etc/shadow, /etc/group

### 11.13 시스템이 부팅할때 특정프로그램을 자동 실행

/etc/rc.d/rc.local 이라는 파일안의 내용이 부팅될때 자동으로 실행된다.

### 11.14 Login 메시지

로컬로 접속하는 경우 → /etc/issue,

네트워크를 경유할 경우 → /etc/issue.net



### 11.15 리눅스 사이트

[www.linux.co.kr](http://www.linux.co.kr)

[www.kldp.org](http://www.kldp.org)

[www.alzzalinux.com](http://www.alzzalinux.com)

[www.linux.sarang.net](http://www.linux.sarang.net)

[www.linux.com](http://www.linux.com)

[www.freshmeat.net](http://www.freshmeat.net)

[www.linux.org](http://www.linux.org)

[www.linuxgazette.com](http://www.linuxgazette.com)

[www.slashdot.org](http://www.slashdot.org)

<http://cyber.woosuk.ac.kr/yoonyc/cyberschoolroom/os/index-os.html>