

본 강의에서 수업자료로 이용되는 저작물은
저작권법 제25조 수업목적 저작물 이용 보상금제도에 의거,
한국복제전송저작권협회와 약정을 체결하고 적법하게 이용하고 있습니다.
약정범위를 초과하는 사용은 저작권법에 저촉될 수 있으므로
수업자료의 재 복제, 대중 공개·공유 및 수업 목적 외의 사용을 금지합니다.

2021. . .

한국복제전송저작권협회



리눅스 명령어

명령어 매뉴얼과 도움말 얻기

* man, apropos

- 어떤 명령어가 있다는 것과 어떤 상황에서 무엇을 확인하고자 할 때 어떤 명령어를 사용해야 한다는 것을 아는데 필수적인 명령어

명령어 매뉴얼과 도움말 얻기 -1

* 사용법

man <명령어>

[space] : 다음페이지

[b] : 이전페이지

/검색어 [Enter] : 이후 페이지 문자검색

?검색어 [Enter] : 이전 페이지 문자검색

[n] : 다음단어로 이동

[q] : 종료

* 영역별 의미

SYNOPSIS : 명령어 사용법

DESCRIPTION : 명령어에 대한 간략한 설명 및 사용 가능한 옵션

명령어 매뉴얼과 도움말 얻기 -2

* apropos – 키워드를 이용한 도움말

예) apropos login

```
[root@nicegood /root]# apropos login
chsh                (1)  - change your login shell
cuserid [getlogin]  (3)  - get user name
dip [diplogin]      (8)  - handle dialup IP connections
faillog             (5)  - Login failure logging file
faillog             (8)  - examine faillog and set login failure limits
getlogin            (3)  - get user name
issue               (5)  - pre-login message and identification file
login               (1)  - sign on
logname             (1)  - print user's login name
nologin             (5)  - prevent non-root users from logging into the system
rlogin              (1)  - remote login
rlogind             (8)  - remote login server
rlogind [in]        (8)  - remote login server
shells              (5)  - pathnames of valid login shells
ssh                 (1)  - OpenSSH secure shell client (remote login program)
ssh [slogin]        (1)  - OpenSSH secure shell client (remote login program)
utmp                (5)  - login records
wtmp [utmp]         (5)  - login records
[root@nicegood /root]#
```

파일 관련 명령어 – ls

* ls (디렉토리 내 파일의 목록을 보기 위해 사용)

```
[tgha@nicegood tgha]$ ls
211_33_41_166_open  cprogram  insi      movie     nessus_source  program
Desktop             database  mail_source music      nsmail         public_html
ah_doc              hacking   mbox      nessus     oper           source
[tgha@nicegood tgha]$
```

* ls 기본 옵션

ls -a	모든 파일 보기
ls -l	자세히 보기(파일의 유형, 허가권, 링크카운트, 파일소유권...)
ls -c	시간 순서대로 정렬해서 보기
ls -d	디렉토리만 보기
ls -f	디스크에 저장된 순서로 보기(정렬 안하기)
ls -i	고유번호(inode)와 함께 보여주기
ls -k	kb 단위로 보여주기
ls -m	파일이름을 가로로 나열하기
ls -r	내림차순 정렬로 보기
ls -t	시간 순서대로 정렬해서 보기
ls -R	하위 디렉토리 안의 파일까지 보여주기
ls --color	파일의 형태에 따라 색깔을 다르게 보여주기
ls -F	파일과 디렉토리를 심볼로 구분하여 보여주기

파일 관련 명령어 – cd

* cd (디렉토리 이동 명령어)

```
[tgha@kcac-linux tgha]$ cd /tmp
[tgha@kcac-linux /tmp]$ pwd
/tmp
[tgha@kcac-linux /tmp]$ cd
[tgha@kcac-linux tgha]$ pwd
/home/tgha
[tgha@kcac-linux tgha]$ cd public_html
[tgha@kcac-linux public_html]$ pwd
/home/tgha/public_html
[tgha@kcac-linux public_html]$ _
```

* cd 기본 옵션

cd	사용자의 홈 디렉토리로 이동
cd ~/	사용자의 홈 디렉토리의 서브디렉토리로 이동
cd -	방금 전에 있던 디렉토리로 이동

디렉토리 관련 명령어 - pwd

현재 디렉토리의 전체 경로를 보여줌

*pwd

파일 관련 명령어 - touch

0(zero) byte 파일 aa 생성

* touch aa

파일 관련 명령어 – cp

* cp (파일 복사 명령어)

```
[root@nicegood public_html]# ls -l
합계 4
-rw-r--r--    1 root    root          2018  4월  6 23:21 test.html
[root@nicegood public_html]# cp test.html hello.html
[root@nicegood public_html]# ls -l
합계 8
-rw-r--r--    1 root    root          2018  4월  6 23:21 hello.html
-rw-r--r--    1 root    root          2018  4월  6 23:21 test.html
[root@nicegood public_html]#
```

* cp 기본 옵션

cp -a	파일의 속성, 링크 정보 등을 유지하면서 복사
cp -b	이미 파일이 존재할 경우 백업 파일 생성
cp -f	기존의 파일을 강제로 지우고 복사
cp -i	복사할 때 물어봄
cp -l	하드링크 형식으로 복사
cp -P	원본파일에 경로와 함께 지정했을 경우 그 경로를 그대로 복사
cp -p	원본파일의 소유주, 그룹, 권한, 시간정보 복사
cp -r	경로와 함께 경로 안의 파일들을 모두 복사

파일 관련 명령어 - mv

* mv (파일명을 변경하거나 다른 곳으로 이동 시키는 명령어)

```
[root@nicegood public_html]# ls -l
합계 4
-rw-r--r--  1 root    root      2018  4월  6 23:21 test.html
[root@nicegood public_html]# mv test.html hello.html
[root@nicegood public_html]# ls -l
합계 4
-rw-r--r--  1 root    root      2018  4월  6 23:21 hello.html
[root@nicegood public_html]# _
```

* mv 기본 옵션

mv -b	백업파일을 만든다.
mv -f	강제로 이동.
mv -i	사용자에게 확인시킨다.
mv -u	업데이트 한다.
mv -v	과정을 자세히 보여준다.

디렉토리 관련 명령어 – mkdir

* 디렉토리를 만들 때 mkdir을 사용한다.

```
[root@nicegood imsil# ls -l
합계 236
drwxrwxr-x    2 root    root      4096  3  28 04:41 kkoc
-rw-rw-r--    1 root    root         0  3  26 07:49 source.c
-r--r--r--    1 root    root    226914  3  26 07:51 source.htm
-rw-r--r--    1 root    root     2018  3  26 07:51 test.html
[root@nicegood imsil# mkdir public_html
[root@nicegood imsil# ls -l
합계 240
drwxrwxr-x    2 root    root      4096  3  28 04:41 kkoc
drwxr-xr-x    2 root    root      4096  4  6 22:59 public_html
-rw-rw-r--    1 root    root         0  3  26 07:49 source.c
-r--r--r--    1 root    root    226914  3  26 07:51 source.htm
-rw-r--r--    1 root    root     2018  3  26 07:51 test.html
[root@nicegood imsil#
```

-p 옵션을 사용하면 하부 디렉토리를 만들수 있다.

* mkdir -p test1/test2

디렉토리 관련 명령어 – rmdir

* rmdir 은 디렉토리를 삭제하는 명령이다.

```
[root@nicegood imsil# ls -l
합계 240
drwxrwxr-x    2 root    root        4096 3월 28 04:41 kkoc
drwxr-xr-x    2 root    root        4096 4월  6 23:19 public_html
-rw-rw-r--    1 root    root           0 3월 26 07:49 source.c
-r--r--r--    1 root    root       226914 3월 26 07:51 source.htm
-rw-r--r--    1 root    root        2018 3월 26 07:51 test.html
[root@nicegood imsil# rmdir public_html/
rmdir: public_html: 디렉토리가 비어있지 않음
[root@nicegood imsil# rm -fr public_html/
[root@nicegood imsil#
```

* 디렉토리가 비어 있어야 한다. 안의 내용을 지우거나 아니면 rm 명령어를 사용할 수 있다.

파일 관련 명령어 – ln

* ln (파일사이에 링크를 만드는 명령)

ln [옵션] [링크시킬 파일이나 폴더] [링크로 만들어질 파일]

* ln 기본 옵션

ln -s	심볼릭 링크
ln -n	파일이 존재하면 겹쳐쓰기를 하지 않는다.
ln -f	파일이 존재해도 겹쳐쓴다.

파일 관련 명령어 – ln

- * 하드 링크(하나의 파일을 여러 곳으로 연결하는데 사용, 동일한 inode 사용)

```
[root@nicegood imsil# ls -l
합계 4
-rw-r--r--    1 root    root          2018  3월 26 07:51 test.html
[root@nicegood imsil# ln test.html index.html
[root@nicegood imsil# ls -il
합계 8
179939 -rw-r--r--    2 root    root          2018  3월 26 07:51 index.html
179939 -rw-r--r--    2 root    root          2018  3월 26 07:51 test.html
```

- * 심볼릭 링크(링크파일을 원본파일로 연결하는 역할)

```
[root@nicegood imsil# ls -l
합계 4
-rw-r--r--    1 root    root          2018  3월 26 07:51 test
[root@nicegood imsil# ln -s test index
[root@nicegood imsil# ls -il
합계 4
179924 lrwxrwxrwx    1 root    root           4 5월 12 22:12 index -> test
179939 -rw-r--r--    1 root    root          2018  3월 26 07:51 test
```

파일 관련 명령어 - ln

- 심볼릭 링크(링크파일을 디렉토리로 연결하는 역할)

* `ln -s /home/ysu1602/a1/a2/a3 myDir`

파일 관련 명령어 – rm

* rm 기본 옵션

rm -r	디렉토리까지 지워버리는 옵션
rm -f	삭제하기 전에 확인과정없이 강제로 지워버리는 옵션
rm -i	사용자에게 정말로 지울 것인지 확인
rm -v	파일 지우는 정보를 자세히 보여줌

파일 관련 명령어 – rm

* rm (불필요한 파일의 삭제)

```
[root@nicegood insil# ls -l
합계 244
-rw-r--r--    1 root    root          2018  3 26 08:05 hello.html
drwxrwxr-x    2 root    root         4096  3 28 04:41 kkoc
drwxrwxr-x    2 root    root         4096  3 26 08:19 public_html
-rw-rw-r--    1 root    root           0  3 26 07:49 source.c
-r--r--r--    1 root    root       226914  3 26 07:51 source.htm
-rw-r--r--    1 root    root          2018  3 26 07:51 test.html
[root@nicegood insil# rm hello.html
rm: 'hello.html'를 지울까요? y
[root@nicegood insil# rm -fr public_html/
[root@nicegood insil# ls -l
합계 236
drwxrwxr-x    2 root    root         4096  3 28 04:41 kkoc
-rw-rw-r--    1 root    root           0  3 26 07:49 source.c
-r--r--r--    1 root    root       226914  3 26 07:51 source.htm
-rw-r--r--    1 root    root          2018  3 26 07:51 test.html
[root@nicegood insil# _
```

파일 관련 명령어 – chmod

* chmod (파일의 사용권한을 조정하는 파일이다.)

```
[root@kcac-linux public_html]# ls -l
total 161344
-rw-r--r--  1 tgha    tgha    165048320 Apr  8 21:54 public_html.tar
[root@kcac-linux public_html]# gzip -lv public_html.tar
public_html.tar:      20.8% -- replaced with public_html.tar.gz
[root@kcac-linux public_html]# gzip -9v public_html.tar
public_html.tar:      21.8% -- replaced with public_html.tar.gz
[root@kcac-linux public_html]#
```

* 파일 권한 보기



r : 읽기 권한
w : 쓰기 권한
x : 실행 권한

파일 관련 명령어 – chmod

* chmod 방법 1

	주인	그룹	다른이
chmod 640 file1	rw-	r--	---
chmod 754 file1	rwX	r-X	r--
chmod 664 file1	rw-	rw-	r--

* chmod 방법 2

u : 소유자
g : 그룹권한
o : 다른 사람
a : 모두

+ : 허가권한 부여
- : 허가권한 제거
= : 해당 권한만 부여

*chmod u+wx file1

*chmod o-x file1

디렉토리 및 하위 디렉토리의 파일까지 한꺼번에 고치려 할 때 -R 옵션을 사용한다.

파일 관련 명령어 – chown

* chown (파일의 소유권을 바꾸는 명령어)

```
[root@nicegood public_html]# ls -l
합계 4
-rw-r--r--  1 bmseo  bmseo      2018  4월  6 23:21 hello.html
[root@nicegood public_html]# chown tgha:tgha hello.html
[root@nicegood public_html]# ls -l
합계 4
-rw-r--r--  1 tgha    tgha      2018  4월  6 23:21 hello.html
[root@nicegood public_html]#
```

- 자기 소유의 파일과 같은 그룹의 파일을 바꿀수 있다.
- root는 모든 파일의 소유권을 바꿀수 있다.

파일 관련 명령어 – cat

* cat (파일의 내용을 보는 명령어)

```
[root@nicegood kkoc]# cat 111.html
11111
22222
[root@nicegood kkoc]# cat 222.html
33333
44444
[root@nicegood kkoc]# cat 111.html 222.html > 333.html
[root@nicegood kkoc]# cat -n 333.html
  1  11111
  2  22222
  3  33333
  4  44444
[root@nicegood kkoc]# cat 111.html 222.html >> 333.html
[root@nicegood kkoc]# cat -b 333.html
  1  11111
  2  22222
  3  33333
  4  44444
  5  11111
  6  22222
  7  33333
  8  44444
[root@nicegood kkoc]#
```


파일 관련 명령어 – head, more

* head (파일 첫 부분을 기준으로 출력), tail(파일 끝 부분)

```
[root@nicegood logs]# tail -3 error_log
[Sat Apr  6 23:20:47 2002] [error] [client 165.246.49.141] File does not exist:
/home/html/scripts/..%2f../winnt/system32/cmd.exe
[Sat Apr  6 23:38:04 2002] [notice] caught SIGTERM, shutting down
[Sat Apr  6 23:38:06 2002] [notice] Apache/1.3.9 (Unix) PHP/4.0.5 configured --
resuming normal operations
[root@nicegood logs]#
```

* more (한 페이지 단위로 넘어가며 텍스트 파일을 보여준다)

```
[root@nicegood logs]# more error_log
[Wed Aug  8 01:34:19 2001] [notice] Apache/1.3.9 (Unix) PHP/4.0.5 configured --
resuming normal operations
[Wed Aug  8 01:37:02 2001] [error] [client 165.194.27.124] File does not exist:
/usr/local/apache/htdocs/default.ida
[Wed Aug  8 02:11:58 2001] [error] [client 165.246.104.180] File does not exist:
/usr/local/apache/htdocs/default.ida
[Wed Aug  8 02:13:01 2001] [error] [client 165.246.95.46] File does not exist: /
--More--(<0%)
```

* less (한페이지 단위로 넘어가며 텍스트 파일을 출력)

- 검색이 가능
- space-bar 한페이지 단위, enter 한행

파일 관련 명령어 – df, wc

* df (파일 시스템 별로 남은 공간을 확인하여 준다.)

```
[tgha@nicegood tgha]$ df -k
Filesystem          1k-blocks      Used Available Use% Mounted on
/dev/hda1           15124868    4849152   9507412   34% /
/dev/hda3            3526204     588820   2758260   18% /user
[tgha@nicegood tgha]$
```

* df 옵션

*df -a 파일시스템을 모두 보여줌

*df -k 용량을 kbyte 단위로 보여줌

* WC (파일안의 글자수, 단어 수, 줄 수를 세어주는 명령어)

```
[root@kcac-linux root]# wc install.log
1722      5697    66145 install.log
```


파일 관련 명령어 – du, sort, file

- * du (디렉토리의 사용량을 출력)
- * sort (텍스트로 된 파일을 정렬해야 할 경우 사용함)
 - *sort aa > aa1
- * file (파일의 종류를 표시 해줌)
 - ASCII, bmp 등

파일 관련 명령어 – find

* **find** (윈도우에서의 찾기, 다양한 옵션으로 조건에 합당한 것을 찾을수 있다)

find [찾을 디렉토리 경로] [찾기옵션] [찾은후 할 작업]

[찾을 디렉토리 경로]

. : 현재 디렉토리

/ : 루트디렉토리 이하(파일시스템 전체)

~ID : 특정ID의 홈디렉토리 이하

[찾기 옵션]

-empty : 비어있는 파일

-gid n : 특정 gid을 갖는 파일

-name : 지정한 형식을 갖는 파일이름

-perm : 특정 허가 모드를 가지고 있는 파일

-user : 특정파일을 소유하고 있는 소유자의 파일

```
# find / -user tgha -name hanbit
```

파일 관련 명령어

touch, which, whatis, whereis

- * touch (비어있는 파일을 만든다)
touch abcde.txt

- * which
(find는 특정 파일을 찾지만, which는 PATH안의 특정 명령어의 위치를 찾아주는 파일이다)
which man

- * whereis
(whereis는 모든 디렉토리를 다 뒤져서 명령어를 찾아준다.)
whereis man

프로세스 명령어 – ps

* ps (실행하고 있는 프로세스를 볼수 있는 명령)

```
[root@nicegood /root]# ps -aux | more
USER          PID %CPU %MEM    USZ    RSS TTY      STAT START   TIME COMMAND
root           1   0.0  0.1  1308     68 ?        S    Apr03   0:05 init [5]
root           2   0.0  0.0      0      0 ?        SW   Apr03   0:00 [kflushd]
root           3   0.0  0.0      0      0 ?        SW   Apr03   0:00 [kupdate]
root           4   0.0  0.0      0      0 ?        SW   Apr03   0:00 [kpiod]
root           5   0.0  0.0      0      0 ?        SW   Apr03   0:00 [kswapd]
root           6   0.0  0.0      0      0 ?        SW<  Apr03   0:00 [mdrecoveryd]
root          61   0.0  0.0      0      0 ?        SW   Apr03   0:00 [khubd]
root          336   0.0  0.5  1368   372 ?        S    Apr03   0:00 syslogd -m 0
```

* ps 기본 옵션

ps -a	다른 사용자의 프로세스 현황도 표시
ps -u	실행한 유저와 실행 시간으로 표시
ps -x	터미널 제어없이 프로세스 현황 표시
ps -e	모든 프로세스 선택하여 표시
ps -f	실행되는 모든 것을 표시
PS -l	긴 포맷으로 출력

프로세스 명령어 – ps

* ps 항목별 내용

항목	내용
USER	프로세스 소유자의 사용자 이름으로 UID
PID	프로그램의 식별번호
%CPU	프로세스가 사용하는 CPU의 백분율
%MEM	프로다음은 ps 명령과 함께 사용할 수 있는 옵션에 대한 설명이다.
VSZ	페이지 단위의 가상 메모리 용량(KB 단위)
RSS	실제 메모리 사용량
TTY	프로세스와 연결된 터미널 포트 ?는 시스템 부팅시 데몬에서 실행되었다는 것을 의미
STAT	현재 프로세스의 상태를 의미 R(Runnable)은 실행대기 상태, S(Sleeping)는 수면상태, D(in Disk wait)는 입출력을 기다리는 인터럽트가 불가능한 상태, T(sTopped)는 멈춰있거나 흔적이 남아있는 상태이고 Z(Zombie)는 완전히 죽어있는 상태
START	프로그램이 시작된 시간
TIME	프로세스가 사용한 CPU 총 사용시간
COMMAND	프로세스의 실행 명령어

프로세스 명령어 – ps

* ps -ef 항목별 내용

항목	내용
UID	소유자의 사용자 이름
PID	프로세스 고유번호
PPID	부모 프로세스의 PID
C	CPU의 사용 및 계획 정보
STIME	프로세스가 시작한 시작
TTY	제어 터미널
TIME	소비한 CPU 시간
COM	명령어 및 인자

프로세스 명령어 – kill

* kill (지정한 프로세스를 종료시킨다)

* kill [-시그널 번호 또는 시그널] PID

```
[tgaha@nicegood tgaha]$ ps -ef | grep sendmail
root      624      1  0 Apr03 ?                00:00:00 sendmail: accepting connections
[tgaha@nicegood tgaha]$ kill -9 624
bash: kill: (624) - Not owner
[tgaha@nicegood tgaha]$ su -
Password:
[root@nicegood /root]# kill -9 624
[root@nicegood /root]# ps -ef | grep sendmail
root      8746   8719  2 05:18 pts/0        00:00:00 grep sendmail
[root@nicegood /root]#
```

프로세스 명령어 – killall

* killall (지정한 이름으로 생성된 모든 프로세스를 종료)

killall [-시그널번호 또는 시그널] 프로세스 이름

```
[root@nicegood /root]# ps -ef | grep httpd
root      8805      1  0 06:08 ?        00:00:00 /usr/local/apache/bin/httpd
nobody    8806    8805  0 06:08 ?        00:00:00 /usr/local/apache/bin/httpd
nobody    8807    8805  0 06:08 ?        00:00:00 /usr/local/apache/bin/httpd
nobody    8808    8805  0 06:08 ?        00:00:00 /usr/local/apache/bin/httpd
nobody    8809    8805  0 06:08 ?        00:00:00 /usr/local/apache/bin/httpd
nobody    8810    8805  0 06:08 ?        00:00:00 /usr/local/apache/bin/httpd
[root@nicegood /root]# killall -KILL httpd
[root@nicegood /root]# ps -ef | grep httpd
root      8816    8719  0 06:08 pts/0    00:00:00 grep httpd
[root@nicegood /root]#
```


프로세스 명령어 – top

- * top (cpu 부하와 프로세스, 메모리사용량 등 시스템 자원에 대한 실시간 정보 제공)

```
4:56pm up 17 days, 23:08, 4 users, load average: 0.00, 0.00, 0.00
82 processes: 80 sleeping, 2 running, 0 zombie, 0 stopped
CPU states: 0.0% user, 0.7% system, 0.0% nice, 99.2% idle
Mem: 190868K av, 185256K used, 5612K free, 0K shrd, 2076K buff
Swap: 1020108K av, 78572K used, 941536K free 126336K cached
```

PID	USER	PRI	NI	SIZE	RSS	SHARE	STAT	%CPU	%MEM	TIME	COMMAND
16873	tgha	15	0	976	976	764	S	0.3	0.5	0:00	top
16908	tgha	17	0	976	976	764	R	0.3	0.5	0:00	top
1	root	8	0	124	72	68	S	0.0	0.0	0:04	init
2	root	8	0	0	0	0	SW	0.0	0.0	0:00	keventd
3	root	9	0	0	0	0	SW	0.0	0.0	0:00	kapm-idled
4	root	9	0	0	0	0	SW	0.0	0.0	0:35	kswapd
5	root	9	0	0	0	0	SW	0.0	0.0	0:00	kreclamd

프로세스 명령어 – pstree

- * **pstree** (실행중인 프로세스 상태를 트리구조로 보여주는 명령으로 이 명령 뒤에 사용자 계정을 지정하면 그 사용자가 실행한 프로세스들을 트리구조로 점검시켜준다.)

pstree [옵션] [pid | user]

```
[root@nicegood root]# pstree
init--+-apmd
      |-atd
      |-bdf flush
      |-bonobo-activati
      |-cannaserver
      |-crond
      |-eth0
      |-gconfd-2
      |-gdm-binary---gdm-binary-+-X
      |                                     ^-gnome-session-+-Xsession
      |                                     ^-ssh-agent
      |-gnome-panel
      |-gnome-settings-
      |-gnome-terminal---bash
      |-gpm
      |-kapmd
      |-keventd
      |-khubd
      |-klogd
      |-ksoftirqd_CPU0
      |-kswapd
      |-kupdated
      |-magicdev
      |-mdrecoveryd
      |-metacity
      |-6*[mingetty]
      |-nautilus---nautilus---3*[nautilus]
      |-nmbd
      |-pam-panel-icon---pam_timestamp_c
      |-portmap
      |-privoxy
      |-rhnp-applet-gui
      |-rpc.statd
      |-2*[sendmail]
      |-smbd
      |-sshd
      |-syslogd
      |-xfs
      ^-xinetd---in.telnetd---login---bash---su---bash---pstree
```

프로세스 명령어

free, vmstat

* free (시스템의 메모리 상태 체크)

```
[root@nicegood root]# free
```

	total	used	free	shared	buffers	cached
Mem:	61540	54684	6856	0	1284	36840
-/+ buffers/cache:		16560	44980			
Swap:	554232	46864	507368			

* Vmstat (가상 메모리 상태를 체크)

```
[root@nicegood root]# vmstat
```

procs				memory			swap		io		system			cpu	
r	b	w	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id
0	0	0	46864	6852	1284	36852	0	1	10	7	223	204	1	0	98

네트워크 명령어 – telnet

* telnet (원격호스트에 접속하여 내 컴퓨터를 사용하는 것처럼 터미널을 열고 실행하고 작업하는 명령어)

* telnet 기본 옵션

telnet -d	디버그 모드로 실행한다.
telnet -n tracefile	추적을 켜고 추적 데이터를 tracefile에 저장한다.
telnet -l user	자동 로그인을 위해 사용자 이름(user)을 원격시스템으로 보낸다. 이 인수는 자동으로 -a 인수를 포함한다.
telnet port	원격시스템에 연결된 포트번호를 나타낸다. 이 인수는 다른 네트워크 프로그램을 지정하는데 사용한다. 지정하지 않으면, telnet이 기본 telnet 포트에 연결된다.

네트워크 명령어 – ftp

* ftp (네트워크에 연결된 컴퓨터 사이에서 파일을 전송하기 위한 명령어)

* ftp 명령 요약

mls	여러 원격 디렉토리의 내용을 나열한다.
mode	파일 전송 모드를 설정한다.
modtime	원격 파일의 마지막 수정 시간을 표시한다.
mput	여러 개의 파일을 원격 컴퓨터로 전송한다.
newer	원격 파일이 로컬 파일보다 최신의 것일 때 원격 파일을 가져온다.
nmap	기본 파일 이름 매핑을 위한 템플릿을 설정한다.
nlist	원격 디렉토리의 내용을 나열한다.
ntrans	기본 파일 이름 매핑을 위한 변환 테이블을 설정한다.
open	원격 FTP 사이트로 연결한다. op라고 쓰기도 한다.
passive	수동 전송 모드를 사용한다.
prompt	여러 개의 명령에 대해 대화식 프롬프트 메시지를 표시하도록 한다.
proxy	대체 연결에서 명령을 발행한다.
put	하나의 파일을 원격 컴퓨터로 전송한다.
pwd	원격 시스템의 작업 디렉토리를 인쇄한다.

네트워크 명령어 – ftp

* ftp 명령 요약

quit	FTP 세션을 종료한다.
quote	임의의 FTP 명령을 전송한다.
recv	파일을 수신한다.
reget	로컬 파일의 끝에서 다시 시작하는 파일을 가져온다.
rstatus	원격 시스템의 상태를 표시한다.
rhelP	원격 서버에서 도움말을 가져온다.
rename	파일의 이름을 변경한다.
reset	큐에 들어간 명령 응답을 삭제한다.
restart	지정된 바이트 계수에서 파일 전송을 다시 시작한다.
rmdir	원격 시스템의 디렉토리를 제거한다.
runique	같은 디렉토리 구조를 갖고 같은 파일 이름을 가진 여러 개의 파일을 검색할 때 수신된 각 파일에 고유 파일 이름을 할당한다.

네트워크 명령어 – ftp

* ftp 명령 요약

runique	같은 디렉토리 구조를 갖고 같은 파일 이름을 가진 여러 개의 파일을 검색할 때 수신된 각 파일에 고유 파일 이름을 할당한다.
send	하나의 파일을 원격 컴퓨터에 전송한다.
site	파일 고유의 명령, umask, idle, chmod, help, group, gpass, newer, minfo 중 하나를 원격 서버에 전송한다.
size	원격 파일의 크기를 표시한다.
status	현재 상태를 표시한다.
struct	파일 전송 구조를 설정한다.
system	원격 시스템 유형을 표시한다.
sunique	같은 파일 이름을 가진 여러 개의 파일을 같은 디렉토리로 보낼 때 송신된 각 파일에 고유한 파일 이름을 할당한다.
tenex	tenex 파일 전송 유형을 설정한다.
tick	전송 진행상황 표시를 토글한다.
trace	패킷 추적을 토글한다.
type	파일 전송 유형을 설정한다.

네트워크 명령어 – ftp

* ftp 명령 요약

user	새로운 사용자 정보를 전송한다.
umask	원격 컴퓨터에서 umask를 가져오거나 설정한다.
verbose	자세한 정보 표시 모드를 토글한다.
?	로컬 도움말 정보를 인쇄한다.

* ftp 기본적인 순서

- ftp 호스트네임/ip address (원격호스트 접속)
- user, passwd (인증과정)
- ls / !ls (현재디렉토리와 원격디렉토리 내용 확인)
- bin/ascii (전송파일 타입 결정)
- hash (마크표시를 하여 전송상태 확인)
- put/get (보내고 받기 명령어)
- quit (접속 끊기)

네트워크 명령어 – ping

- ping (원격시스템에 제대로 연결되어 있는지 상태를 확인)

#ping [hostname 또는 ip address]

```
C:\W>ping nicegood.zdns.net

Pinging nicegood.zdns.net [165.246.142.29] with 32 bytes of data:

Reply from 165.246.142.29: bytes=32 time<10ms TTL=242
Reply from 165.246.142.29: bytes=32 time=20ms TTL=242
Reply from 165.246.142.29: bytes=32 time=10ms TTL=242
Reply from 165.246.142.29: bytes=32 time=10ms TTL=242

Ping statistics for 165.246.142.29:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 20ms, Average = 10ms
```

* ping 옵션

ping -s	ping 테스트시에 사용할 패킷사이즈 크기 설정 옵션
ping -q	ping 테스트결과를 지속적으로 보여주지 않고 종합 결과만을 보여주는 옵션
ping -i	ping 테스트시 사용할 interval을 설정 즉, 지연시간을 의미함
ping -b	ping 테스트를 하는 서버와 동일한 네트워크에 있는 모든 호스트로 패킷을보냄
ping -c	ping 테스트시에 보낼 패킷수를 지정해주는 옵션
ping -t	ping 테스트시 연속하여 멈추지 않고 실행하는 옵션

네트워크 명령어 – finger

- finger (시스템의 특정 사용자에게 대한 정보를 알아보는 명령어)

finger username@domain

```
C:\W>finger tgha@nicegood.zdns.net

[nicegood.zdns.net]
Login: tgha                Name: (null)
Directory: /home/tgha      Shell: /bin/bash
On since Mon May 13 08:18 (KST) on pts/1 from 211.252.151.3
    23 seconds idle
Mail forwarded to tghamath@hanmail.net
No mail.
No Plan.
```

네트워크 명령어 – traceroute

- * traceroute (원격시스템까지 가는 경로를 조사하여 화면에 출력하는 명령어)
#traceroute [테스트할 주소]
- * whois (특정 도메인의 정보나 특정 ip주소를 사용하는 기관 정보를 조회할수 있다)

네트워크 명령어 – whois

whois 서버의 종류

whois.arin.net : 최상위 도메인 정보 제공

whois.apnic.net : 아시아 지역 도메인 정보 제공

whois.ripe.net : 유럽지역의 도메인 정보 제공

whois.krnic.net : 국내 도메인 정보 제공

국내 도메인 사용기관 정보 조회 방법

리눅스셸에서의 조회 : `whois -h whois.krnic.co.kr ip_address`

웹에서의 조회 : <http://whois.nic.or.kr>

국외 도메인 사용기관 정보 조회 방법

리눅스셸에서의 조회 : `whois -h whois.internic.net ip_address`

웹에서의 조회 : <http://www.netsol.com/cgi-bin/whois/whois>

네트워크 명령어 – netstat

* netstat (자세한 라우팅 테이블 정보와 네트워크 상태 체크 도구)
netstat -nr

* netstat 기본 옵션

netstat -a	--all과 같으며 listen되는 소켓 정보와 listen되지 않는 소켓 정보 모두를 출력
netstat -n	--numeric과 같으며 10진수의 수치 정보로 결과를 출력
netstat -r	--route와 같으며 설정된 라우팅 정보를 출력
netstat -p	--program과 같으며 실행되고 있는 각 프로그램과 PID정보를 출력
netstat -i	--interface=iface과 같으며 모든 네트워크인터페이스 정보를 출력 또는 특정 네트워크 인터페이스를 지정할 수도 있음
netstat -c	--continuous과 같으며 netstat 결과를 연속적으로 출력
netstat -l	--listening과 같으며 현재 listen 되고 있는 소켓 정보를 출력
netstat -s	--statistics과 같으며 각 프로토콜에 대한 통계 정보를 출력

네트워크 명령어 – netstat

- * netstat 예제 (atp 옵션을 사용하여 현재 응답할 수 있는 포트번호와 각 데몬들, 그리고 그 포트를 사용하는 프로그램에 대한 정보를 상세히 검토)

```
[root@nicegood root]# netstat -atp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 *:1024                 *:*                     LISTEN      514/rpc.statd
tcp        0      0 localhost.localdom:1031 *:*                     LISTEN      1306/xinetd
tcp        0      0 *:netbios-ssn          *:*                     LISTEN      1721/smbd
tcp        0      0 *:sunrpc                *:*                     LISTEN      495/portmap
tcp        0      0 *:x11                   *:*                     LISTEN      1119/X
tcp        0      0 *:ftp                   *:*                     LISTEN      1306/xinetd
tcp        0      0 localhost.localdom:8118 *:*                     LISTEN      701/privoxy
tcp        0      0 *:ssh                   *:*                     LISTEN      643/sshd
tcp        0      0 *:telnet                 *:*                     LISTEN      1306/xinetd
tcp        0      0 localhost.localdom:smtp *:*                     LISTEN      681/sendmail: accep
tcp        0    158 nicegood.zdns.ne:telnet 211.191.217.219:1058   ESTABLISHED 3603/in.telnetd: 21
```

압축 명령어 – tar

* **tar** (여러개의 파일을 하나로 묶을수 있으며 묶여진 파일을 다시 풀수 있다)

tar [옵션] [파일이름] [대상 디렉토리 및 파일]

*묶는 작업 : tar cvf [압축파일.tar] [압축대상파일 및 디렉토리]

*확인 작업 : tar tvf [압축파일.tar]

*푸는 작업 : tar xvf [압축파일.tar]

* 파일 묶기

```
[tg@kac-linux public_html]$ ls -l
total 24
-rwxrwxr-x  1 tg  tg  7831 Apr 10 22:51 crl.htm
-rwxrwxr-x  1 tg  tg  1731 Apr 10 22:51 index.html
-rwxrwxr-x  1 tg  tg 11701 Apr 10 22:52 main.html
[tg@kac-linux public_html]$ tar cvf public_html.tar *
crl.htm
index.html
main.html
[tg@kac-linux public_html]$
```


압축 명령어 – tar

* 파일 내용 확인

```
[tgha@kcac-linux public_html]$ ls -l
total 32
-rw-rw-r-- 1 tgha tgha 30720 Apr 10 22:52 public_html.tar
[tgha@kcac-linux public_html]$ tar tvf public_html.tar
-rwxrwxr-x tgha/tgha 7831 2002-04-10 22:51:56 crl.htm
-rwxrwxr-x tgha/tgha 1731 2002-04-10 22:51:47 index.html
-rwxrwxr-x tgha/tgha 11701 2002-04-10 22:52:13 main.html
[tgha@kcac-linux public_html]$
```

* 파일 풀기

```
[tgha@kcac-linux imsil]$ ls -l
total 14820
-rw-rw-r-- 1 tgha tgha 15155200 Apr 10 17:00 php-4.0.5.tar
[tgha@kcac-linux imsil]$ tar xvf php-4.0.5.tar _
```

압축 명령어 - gzip

* gzip (압축 명령어)

압축하기 : gzip [압축할 파일명]

압축풀기 : gzip -d [압축된 파일명]

```
[tgha@kcac-linux public_html]$ ls -l
total 4
-rw-rw-r-- 1 tgha tgha 3904 Apr 11 00:14 public_html.tar
[tgha@kcac-linux public_html]$ gzip public_html.tar
[tgha@kcac-linux public_html]$ ls -l
total 4
-rw-rw-r-- 1 tgha tgha 3943 Apr 11 00:14 public_html.tar.gz
[tgha@kcac-linux public_html]$
```