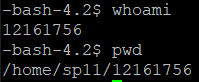
5. Homework

5.1 Do following steps.

1) Log in to a Linux server. Find your login directory with "pwd". Find your user ID with "whoami".

$ pwd

$ whoami



**$ pwd : 현재 디렉토리를 보여준다. (present working directory)**

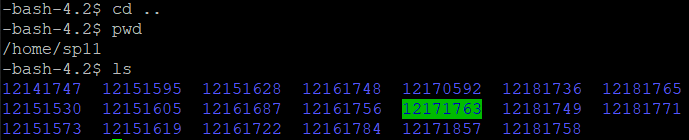
**$ whoami : 자신의 user name을 보여준다.**

2) Go up the file tree with "..", display the current location with "pwd", and find other students directory name with "ls".

$ cd ..

$ pwd

$ ls



**$ cd .. : 부모 디렉토리로 이동한다. (change directory + parent directory)**

**$ pwd : /home/sp11/12161756 -> /home/sp11로 이동한 것을 확인할 수 있다.**

**$ ls : 현재 디렉토리에 있는 파일과 디렉토리명을 출력한다. (list)**

3) Pick one of the student IDs (suppose it was 12345) and try to enter his/her directory with "cd".

$ cd 12345

What happens?



**$ cd 12141747 : 12141747 디렉토리로 이동하려하지만 거부당했다.**

4) Use "ls -l" to see the file permission of the directory you tried to enter.

$ ls -l

............

drwx------ 2 12345 12345 4096 Feb 27 11:47 12345

............

d: this file is a directory

rwx: the owner can read/write/execute this file

--- : the group member has no read/write/execute permission

--- : other people has no read/write/execute permission

2 : there are two links to this file

12345 : the user ID of the owner of this file

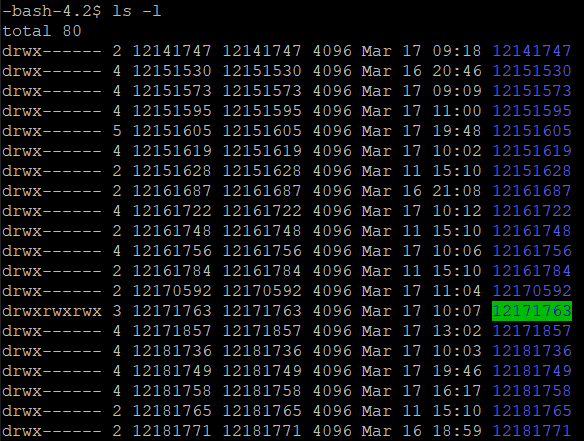
12345 : the group ID of of the owner of this file

4096 : the byte size of this file

Feb 27 11:47 : the creation time or last modification time of this file

12345 : file name

Explain why you couldn't enter the directory you picked.



**$ ls -l : 현재 디렉토리에 있는 파일과 디렉토리의 상세 정보를 출력한다.**

**= 12141747 디렉토리에 접근할 수 없었던 이유는 해당 디렉토리의 권한이 ‘drwx------‘으로 소유자 이외에는 읽기/쓰기/실행 권한이 없기 때문이다.**

5) Find out your current location again with "pwd".

$ pwd



**$ pwd : 현재 디렉토리를 확인한다.**

6) Go to the root directory with "cd /" and make sure you are really at the top directory with "pwd".

$ cd /

$ pwd



**$ cd / : root 디렉토리로 이동한다. (cd + root directory[/])**

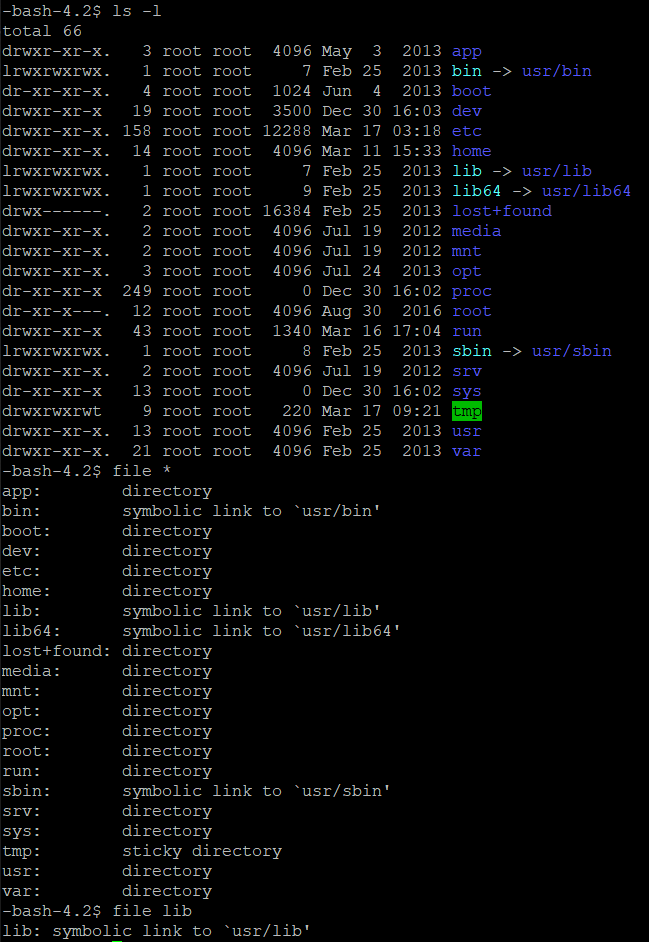
**$ pwd : root 디렉토리[/]로 이동한 것을 확인할 수 있다.**

7) How many files you have in the root directory? Some of them are not directory files. Find them with "ls -l". Use "file" command to see more detailed info.

$ ls -l

$ file \*

$ file lib



**$ ls -l : 현재 디렉토리안에 있는 디렉토리와 파일의 상세 정보를 출력한다.**

**$ file \* : 모든 파일의 타입을 보여준다. (file + 와일드 카드[\*])**

**$ file lib : lib이라는 파일의 타입을 보여준다.**

**= 21개의 파일이 있음을 확인할 수 있다.**

8) \* is a wild card character meaning it will be replaced by all file names in the current directory.

$ file \*

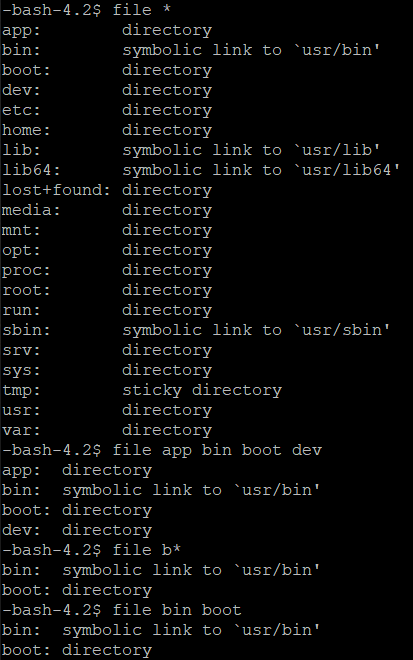
is same as

$ file app bin boot dev .... (all file names in the current directory)

$ file b\*

is same as

$ file bin boot



**$ file \* : 모든 파일들의 타입을 확인한다.**

**$ file app bin boot dev : 현재 디렉토리에 있는 app, bin, boot, dev의 파일 타입을 한 번에 확인한다.**

**$ file b\* : b로 시작하는 파일들의 타입을 확인한다.**

**$ file bin boot : bin과 boot 파일의 타입을 확인한다.**

**= b로 시작하는 파일들이 현재 디렉토리에서는 bin과 boot만 존재하기에 ‘file b\*’과 ‘file bin boot’의 실행 결과가 동일하다.**

9) If you want to go back to your login directory (suppose it was /home/sp1/12345), you can cd with absolute path, cd with relative path, or just "cd".

cd /home/sp1/12345 -- go to absolute path "/home/sp1/12345"

cd home/sp1/12345 -- go to "home" in the current directory and go to "sp1"

and then to "12345", that is go to relative directory

"home/sp1/12345"

cd -- go to the login directory of the current user

Use one of above methods to go back to your login directory (or home directory).



**$ cd : 로그인 디렉토리로 이동한다.**

**= 로그인 디렉토리로 이동하는 방법은 세 가지가 있다. 1. 절대경로 2. 상대경로 3. 그냥 cd입력, 이중 세 번째 방법을 택했다.**

10) Confirm your current location with "pwd".



**= 현재 디렉토리가 루트에서 로그인 디렉토리로 이동한 것을 확인할 수 있다.**

11) List all files in your directory with "ls" command.



**= 현재 디렉토리에 파일이 존재하지 않아 (숨김파일 제외) 아무것도 출력되지 않는다.**

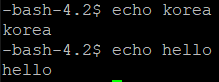
12) Try "echo" command.

$ echo korea

korea

$ echo hello

hello



**$ echo korea : ‘korea’를 출력한다.**

**$ echo hello : ‘hello’를 출력한다.**

11) Try "echo" with ">" symbol. ">" is called "standard output redirection".

$ echo hello > f1

">" sends the result to "f1" so there is no output shown in the terminal but you will have f1 in the current directory.

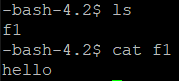


**$ echo hello > f1 : ‘echo hello’결과를 f1 파일에 저장한다.**

12) Do "ls" to see you can find f1 in the current directory. Show its content with "cat".

$ ls

$ cat f1



**$ ls : 파일이 생성된 것을 확인할 수 있다.**

**$ cat f1 : f1 파일의 내용을 확인한다.**

**= standard output redirection을 이용하여 f1에 ‘echo hello’의 결과인 ‘hello’를 저장한 것을 확인할 수 있다.**

13) Make a directory, d1, with "mkdir".

$ mkdir d1



**$ mkdir d1 : d1이라는 디렉토리를 생성한다.**

14) Copy f1 into directory d1.

$ cp f1 d1 -- make a copy of f1 inside directory d1 under the same name.

$ cd d1

$ ls

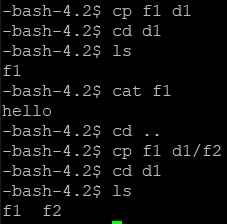
$ cat f1

$ cd ..

$ cp f1 d1/f2 -- make a copy of f1 inside d1 under another name f2

$ cd d1

$ ls



**$ cp f1 d1 : d1 디렉토리 안에 f1의 복사본(이름 똑같이)을 생성한다.**

**$ cd d1 : d1 디렉토리로 이동한다.**

**$ ls : 파일 목록을 출력하여 f1이 생성된 것을 확인한다.**

**$ cat f1 : f1의 내용으로 ‘hello’가 출력되는 것을 확인할 수 있다.**

**$ cd .. : 부모 디렉토리로 이동한다.**

**$ cp f1 d1/f2 : f1의 복사본을 d1 디렉토리안에 ‘f2’의 이름으로 생성한다.**

**$ cd d1 : d1 디렉토리로 이동한다.**

**$ ls : f2가 생성된 것을 확인할 수 있다.**

5.2 Do followings and explain what happens and why.

$ cd

$ ls

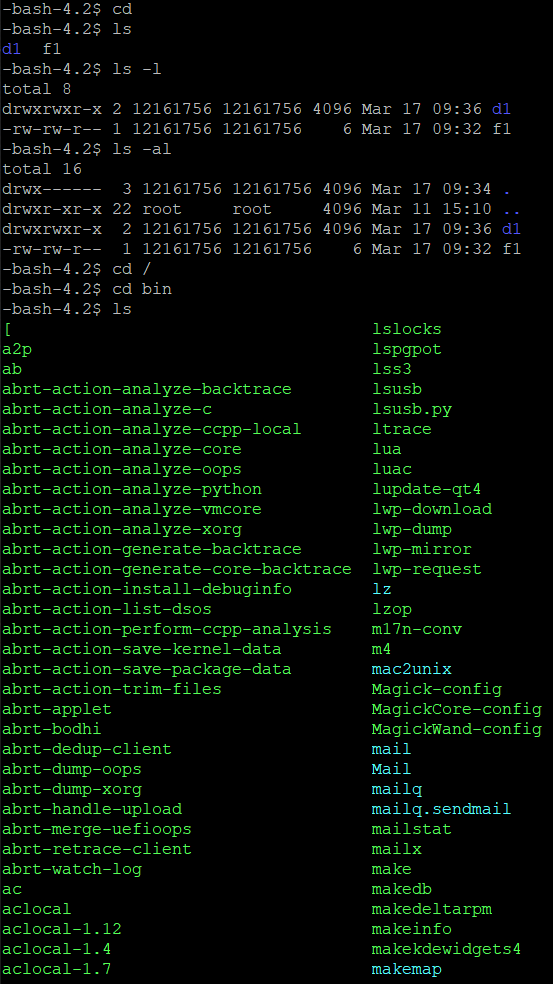
$ ls –l

$ ls –al

$ cd /

$ cd bin

$ ls



**$ cd : 로그인 디렉토리로 이동한다.**

**$ ls : 현재 디렉토리(로그인 디렉토리)에 존재하는 파일 및 디렉토리명을 출력한다.**

**$ ls –l : 현재 디렉토리에 존재하는 파일 및 디렉토리의 상세 정보를 출력한다.**

**$ ls –al : 현재 디렉토리에 존재하는 숨겨진 것 포함 모든 파일 및 디렉토리의 상세 정보를 출력한다. 현재 디렉토리[.]와 부모 디렉토리[..]가 포함된 것을 확인할 수 있다.**

**$ cd / : root 디렉토리로 이동한다.**

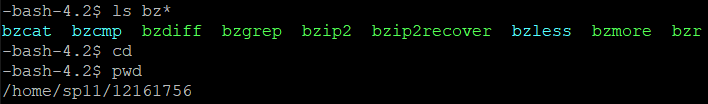
**$ cd bin : bin 디렉토리로 이동한다.**

**$ ls : 현재 디렉토리(/bin)에 존재하는 파일 및 디렉토리명을 출력한다. (일부 생략)**

$ ls bz\*

$ cd

$ pwd

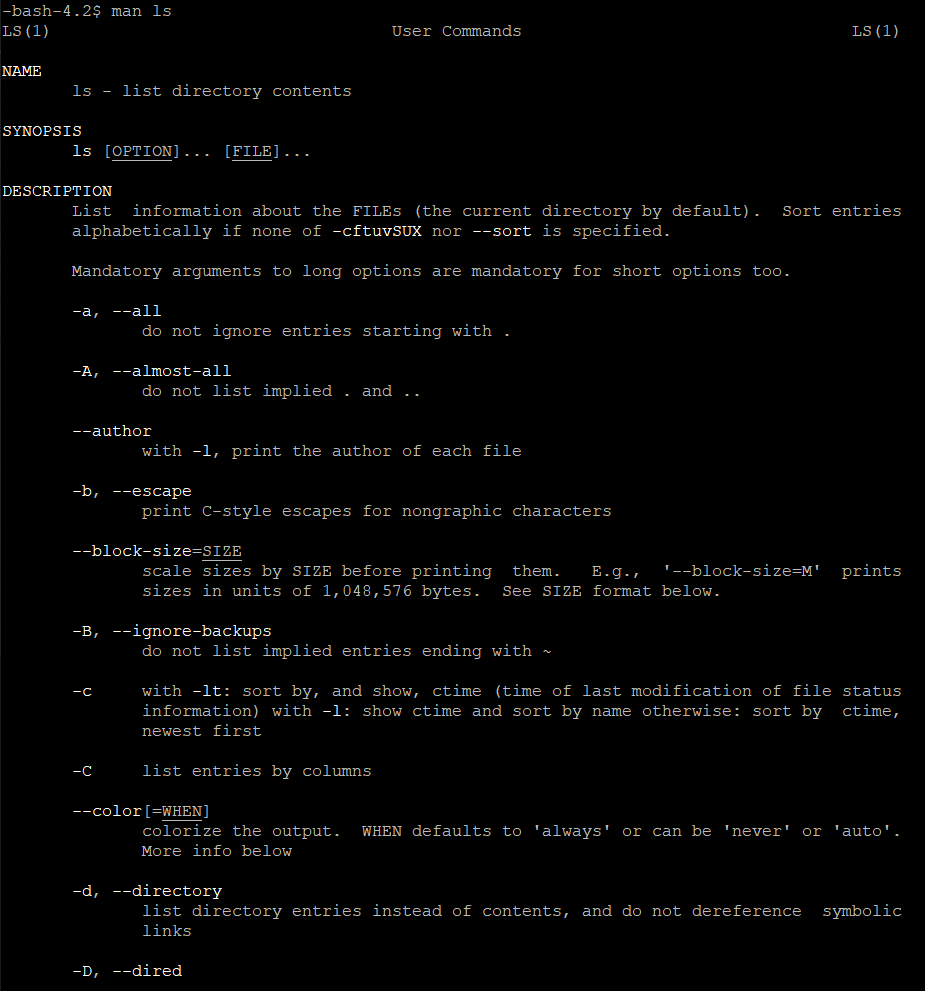


**$ ls bz\* : ‘bz’로 시작하는 파일 및 디렉토리명을 출력한다.**

**$ cd : 로그인 디렉토리로 이동한다.**

**$ pwd : 로그인 디렉토리로 이동한 것을 확인할 수 있다.**

$ man ls



**$ man ls : 유저 명령어 관련 메뉴얼이 출력된다. (일부 생략)**

$ echo hello

$ echo hello > f4

$ ls

$ cp f4 f2

$ cat f4

$ cat f2

$ cat f2 > f3

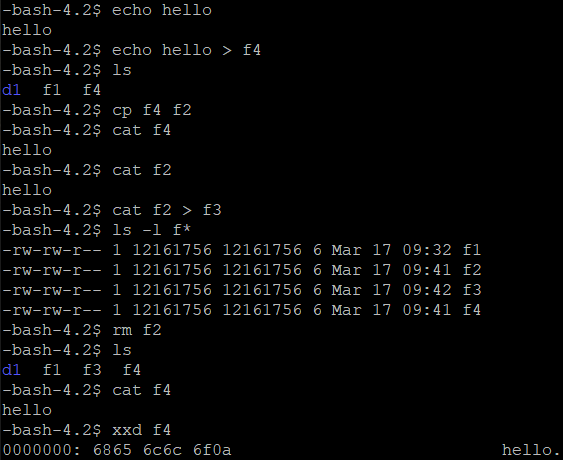
$ ls –l f\*

$ rm f2

$ ls

$ cat f4

$ xxd f4



**$ echo hello : ‘hello’를 출력한다.**

**$ echo hello > f4 : echo hello의 결과[hello]를 f4 파일의 내용으로 기록한다.**

**$ ls : f4가 생성된 것을 확인할 수 있다.**

**$ cp f4 f2 : f4의 복사본 f2를 생성한다.**

**$ cat f4 : f4의 내용이 ‘hello’로 출력된다.**

**$ cat f2 : f2의 내용이 f4와 동일하게 출력되는 것을 확인할 수 있다.**

**$ cat f2 > f3 : f2 내용을 f3에 입력한다.**

**$ ls –l f\* : f로 시작하는 파일 및 디렉토리의 상세 정보를 출력한다.**

**$ rm f2 : f2 파일을 제거한다.**

**$ ls : f2가 제거된 것을 확인할 수 있다.**

**$ cat f4 : f4 내용을 출력한다.**

**$ xxd f4 : f4의 내용을 16진수로 출력한다.**

$ mkdir d2

$ cp f4 d2

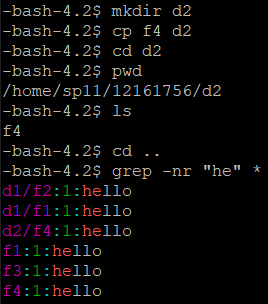
$ cd d2

$ pwd

$ ls

$ cd ..

$ grep –nr "he" \*



**$ mkdir d2 : d2라는 디렉토리를 생성한다.**

**$ cp f4 d2 : f4의 복사본을 d2 하위에 생성한다.**

**$ cd d2 : d2로 이동한다.**

**$ pwd : d2로 이동한 것을 확인할 수 있다.**

**$ ls : d2 안에 f4가 생성된 것을 확인할 수 있다.**

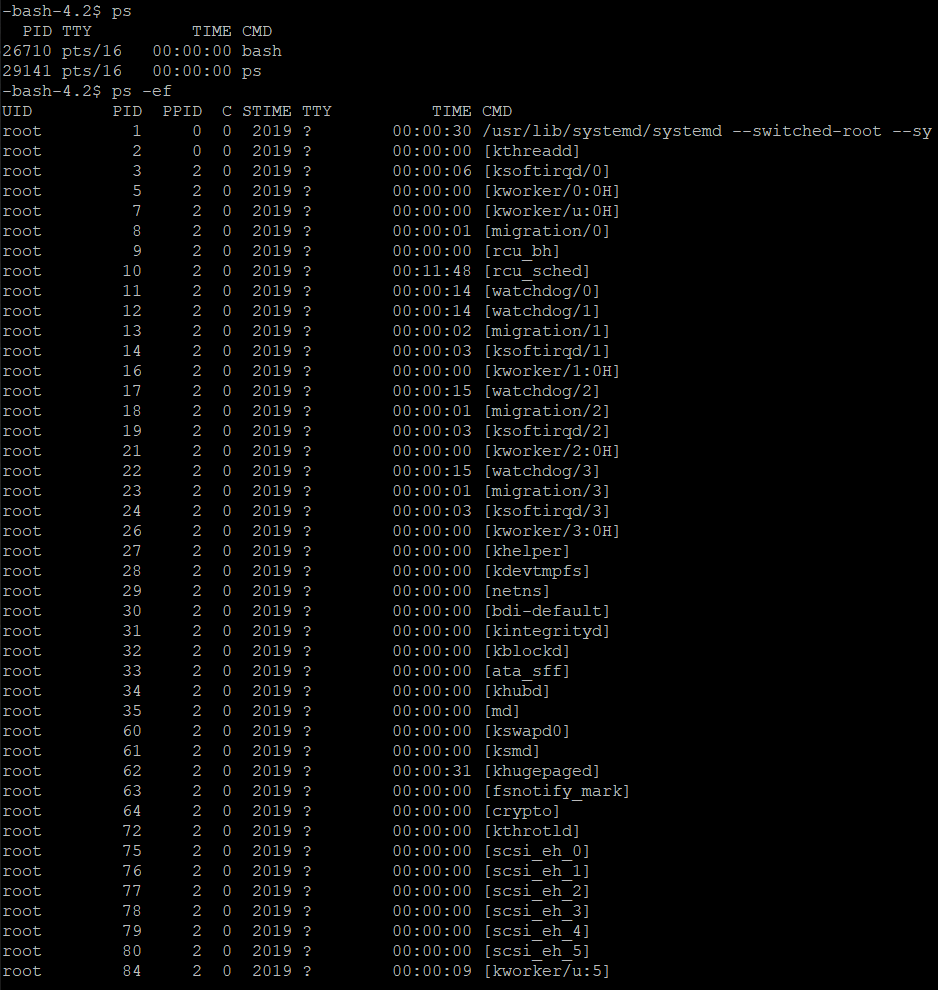
**$ cd .. : 다시 이전 디렉토리로 이동한다.**

**$ grep –nr "he" \* : ‘he’라는 문자열이 포함된 파일과 문자열 번호, 해당 문자열을 출력한다.**

**= [grep] : 문자열 검색, [-n] : 줄 번호도 같이 출력 (line number), [-r] : 모든 하위 디렉토리 안의 파일에 대해서도 수행 (recursively)**

$ ps

$ ps –ef

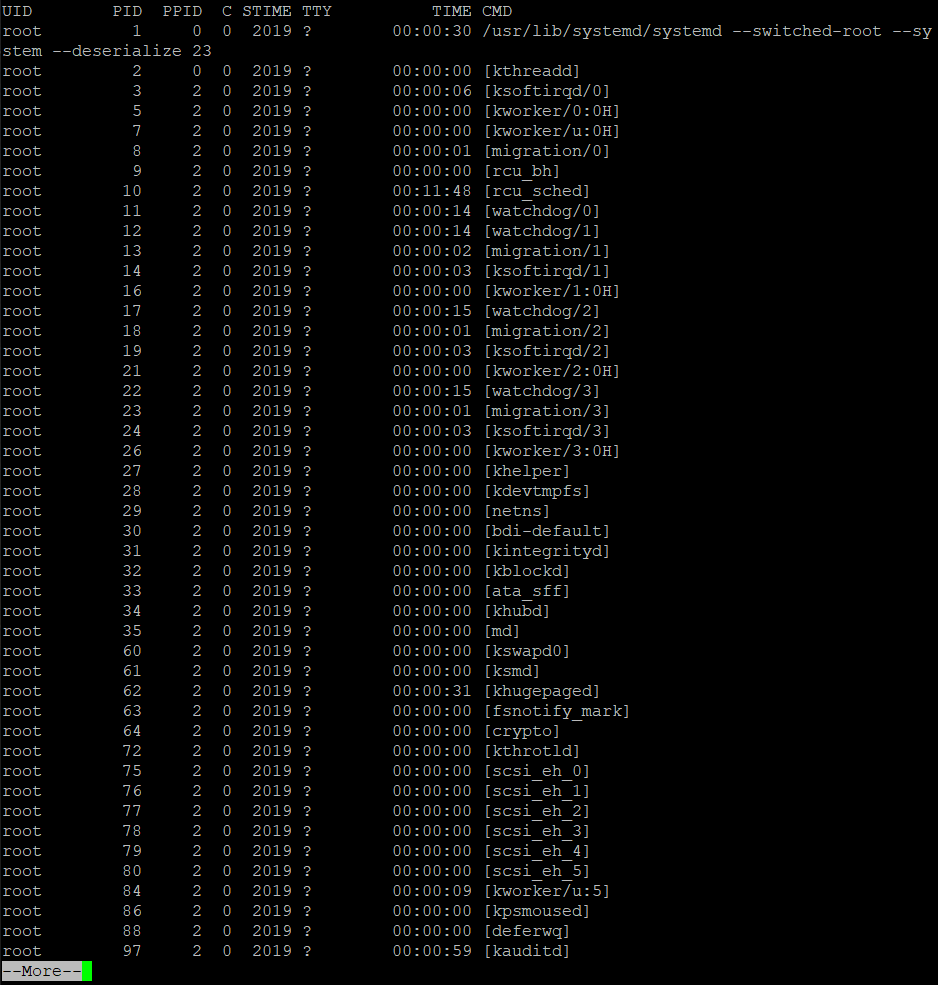


**$ ps : 현재 사용자의 프로세스를 보여준다.**

**$ ps –ef : 모든 사용자의 모든 프로세스를 보여준다. (일부 생략)**

$ ps –ef | more

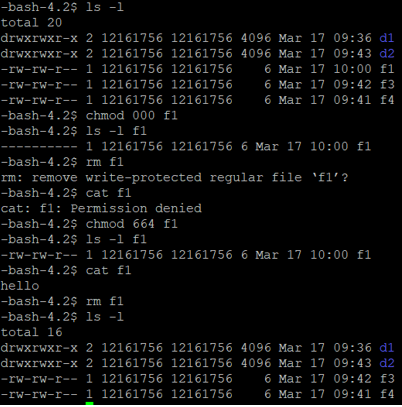




**$ ps –ef | more : pipe 기호[|] 전 명령어의 출력을 기호 후의 명령어의 입력으로 전달하여, 긴 ps -ef의 결과를 more을 통해 한 화면씩 분할하여 출력해주고 있다. space로 다음 내용을 확인하며 q를 눌러 빠져나온다.**

5.3 Find out the usage of following commands. Use them in a Linux server and explain what happens.

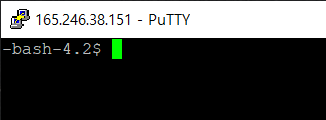
chmod



**$ chmod [옵션] [모드] [파일명] : 파일이나 디렉토리의 허가권을 변경한다.**

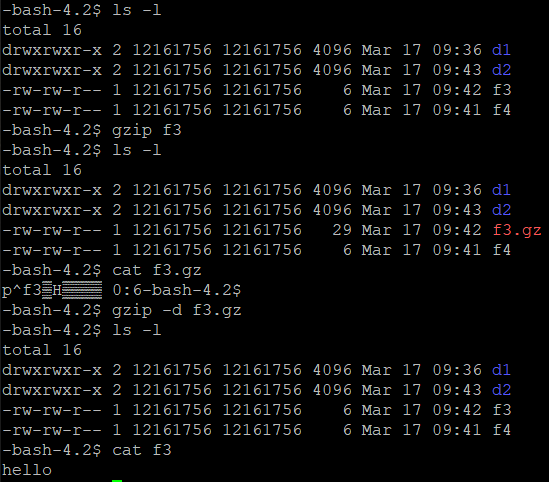
**= f1의 -rw-rw-r— [664] 허가권을 ---------- [000]으로 변경한 모습이며 이후 시도한 rm과 cat도 다 거절되는 모습이다. 다시 [664]로 변경한 이후 cat와 rm 모두 정상 실행되었다.**

clear



**$ clear : 입출력창에 출력된 모든 내용을 지운다.**

gzip



**$ gzip [옵션] [파일명] : 파일을 압축, 압축해제 한다.**

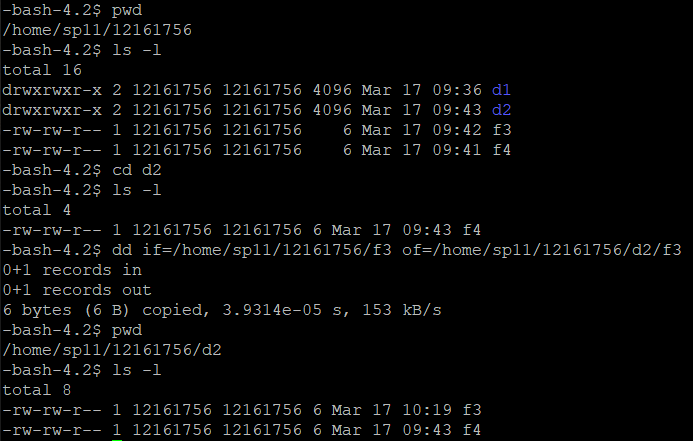
**= f3을 압축하자 f3.gz가 생성되는 것을 확인할 수 있다. [-d] (decompress) 옵션을 이용해 압축을 해제하자 다시 f3이 생성된 것을 확인할 수 있다. 기존 파일의 타입과 용량의 따라 압축 후 용량이 오히려 증가할 수 있음을 보여주고 있다.**

date



**$ date : 현재 날짜와 시간을 출력한다. (기본옵션)**

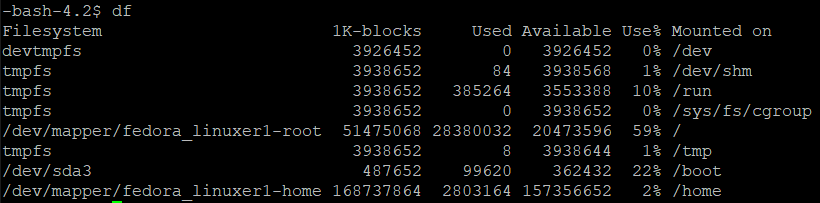
dd



**$ dd if=[원본파일경로] of=[복사본이 생성될 경로] : if에 작성한 파일을 of에 작성한 경로로 복사한다.**

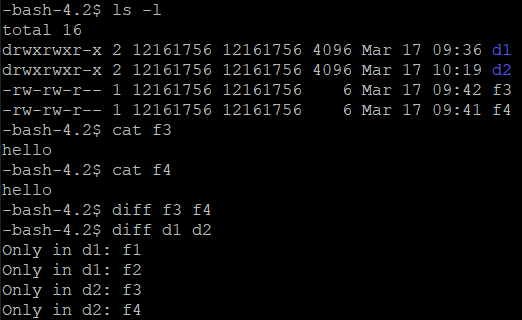
**= /home/sp11/12161756/f3이 /home/sp11/12161756/d2/f3으로 복사된 것을 확인할 수 있다.**

df



**$ df : 파일시스템 디스크 공간의 사용량을 출력한다.**

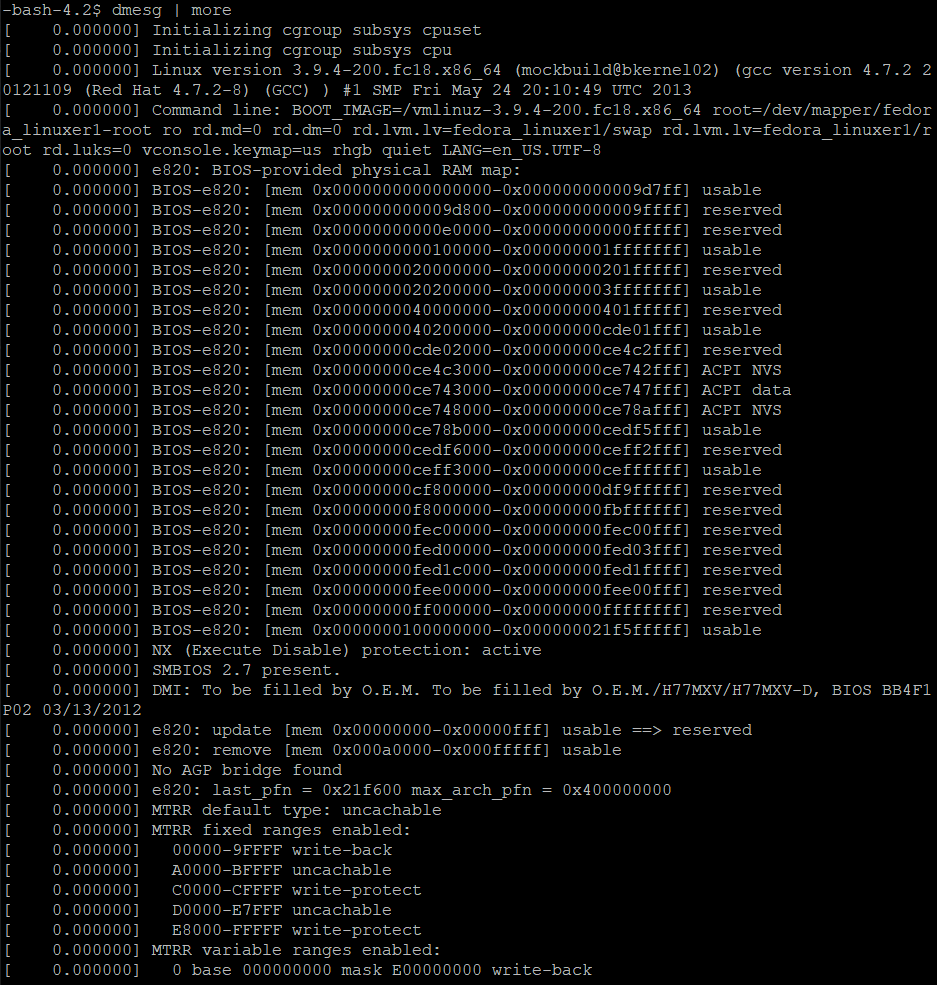
diff



**$ diff [파일명1] [파일명2] : 두 파일이나 디렉토리 간 차이를 출력한다.**

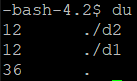
**= f3과 f4는 차이가 없어 출력된 내용이 없다. d1과 d2에 대해서는 차이를 출력해주고 있다. d1에만 f1과 f2 파일이 있고 d2에만은 f3과 f4 파일이 있다.**

dmesg



**$ dmesg : 커널 링 버퍼를 출력한다. 시스템이 부팅할 때마다 로딩되는 시스템 정보의 로그 메시지를 보여준다. (일부 생략)**

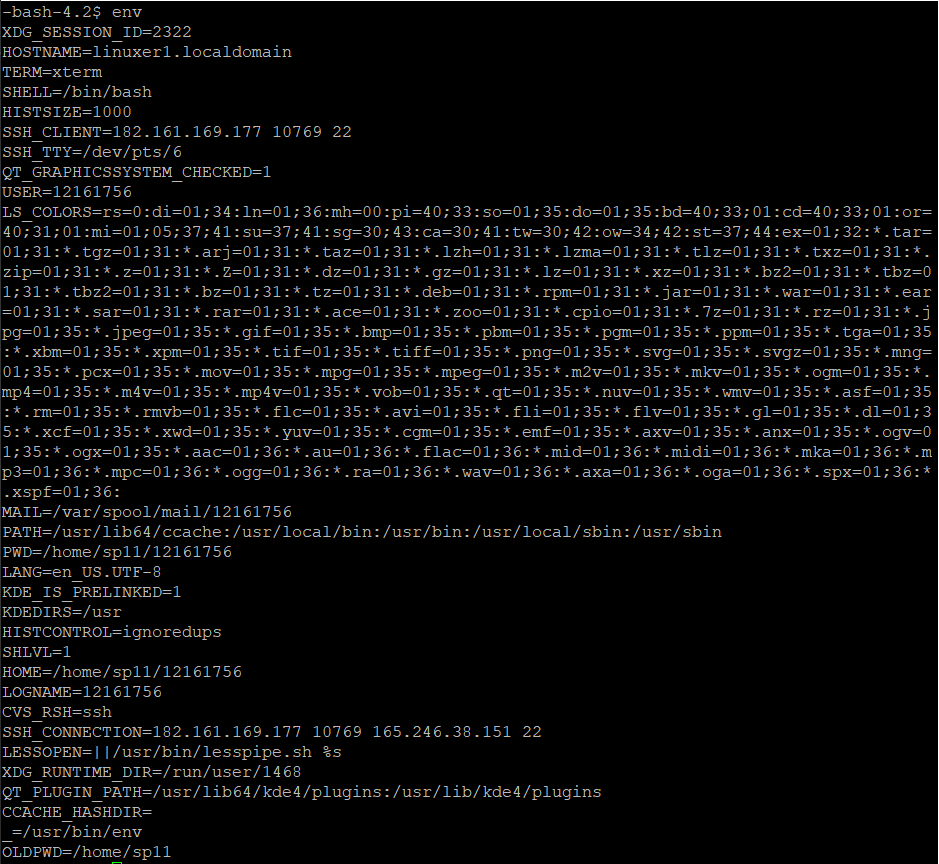
du



**$ du : 현재 디렉터리의 디스크 사용량을 출력한다.**

**= 각 디렉토리별 디스크 사용량이 MB 단위로 출력되었다.**

env



**$ env : 현재 지정되어 있는 환경 변수 값을 확인한다.**

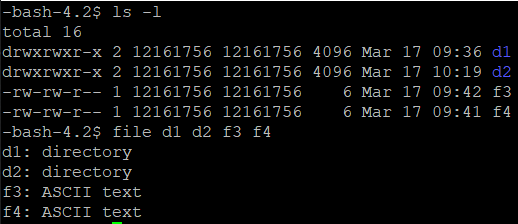
exit



**$ exit : 명령행을 종료한다.**

**= logout된 것을 확인할 수 있다.**

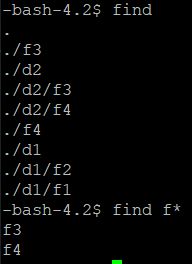
file



**$ file [파일or디렉토리명] : 파일 타입을 확인한다.**

**= d1과 d2는 디렉토리, f3과 f4는 ASCII text 파일로 확인된다.**

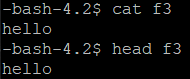
find



**$ find [조건] : 파일을 찾는다.**

**= [find f\*]를 입력하자 f로 시작하는 파일명이 출력되었다.**

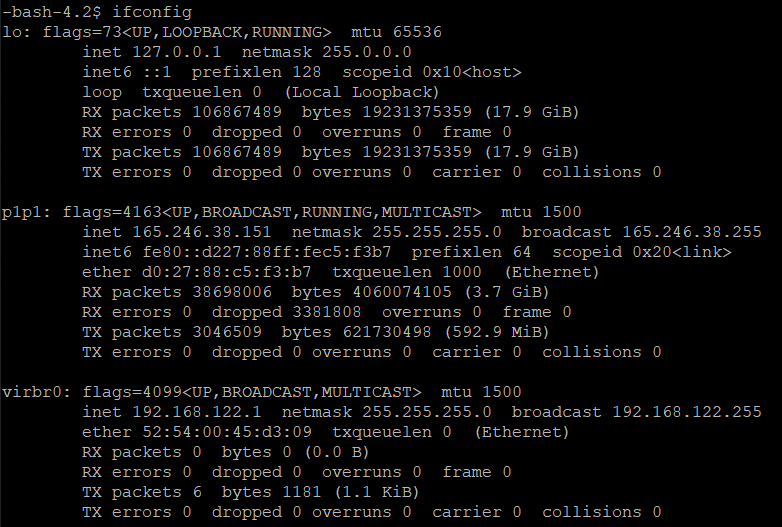
head



**$ head [파일명] : 파일 내용의 위에서부터 10줄(기본값)을 출력한다.**

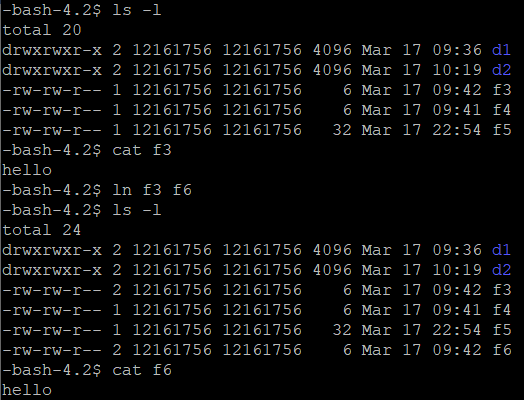
**= f3의 내용은 한 줄뿐이 없어 hello만 출력된 모습이다.**

ifconfig



**$ ifconfig : 이더넷 카드와 네트워크 환경을 확인한다.**

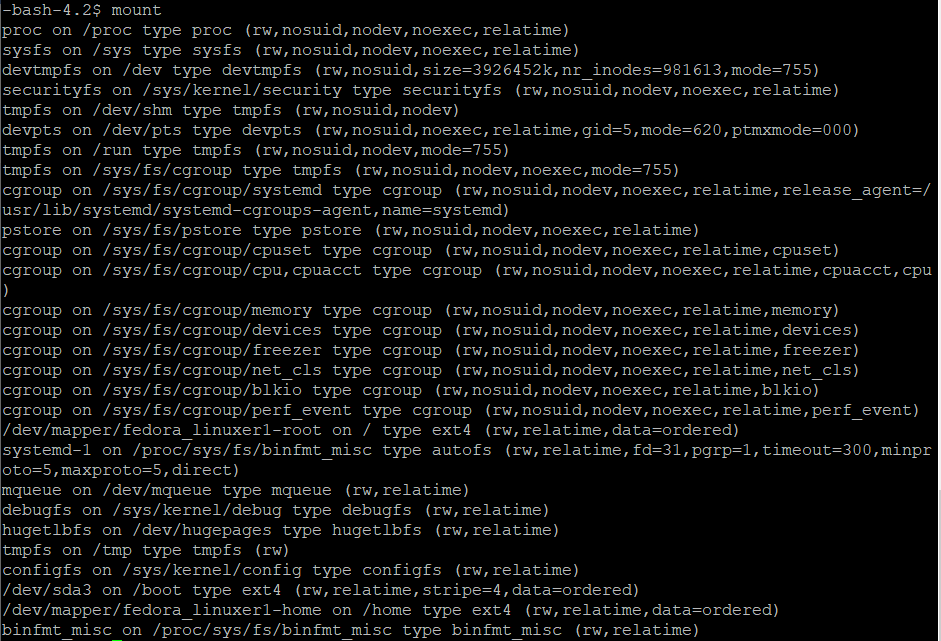
ln



**$ ln [링크대상] [만들링크명] : 파일의 링크를 생성한다.**

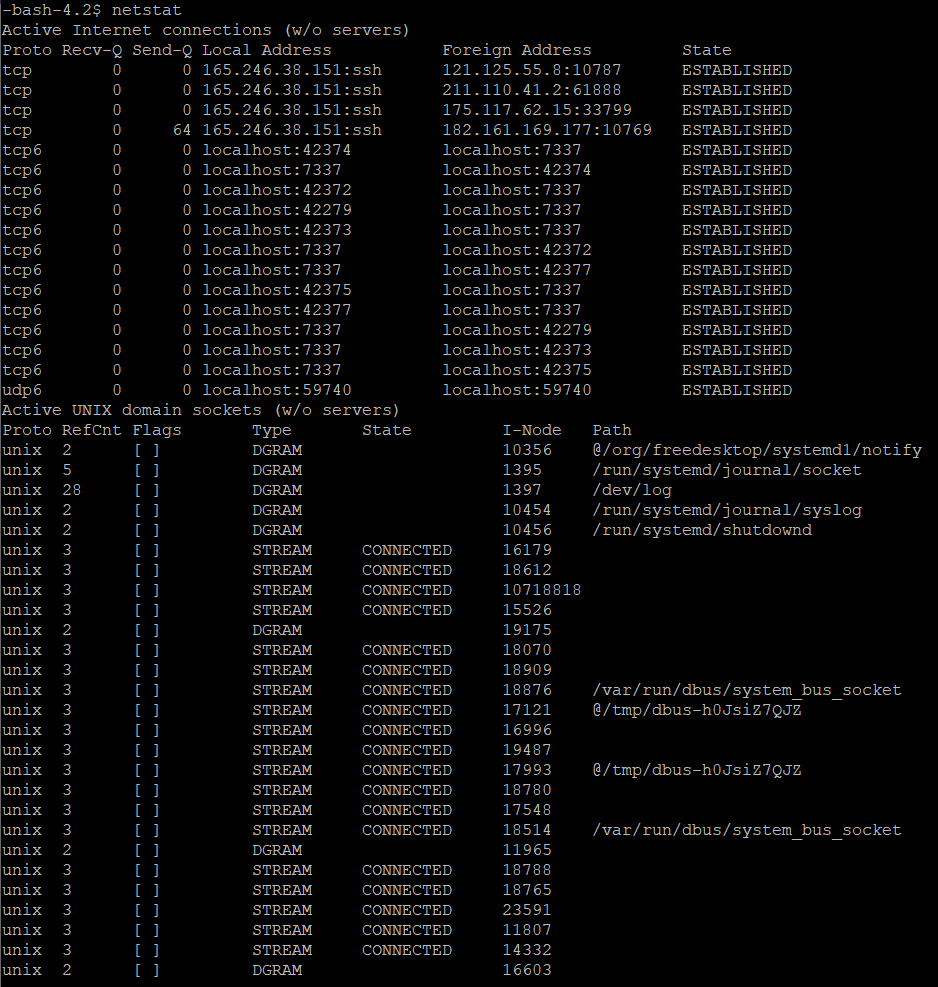
**= [ln f3 f6]을 통해 f3의 링크인 f6을 생성하였고 f6의 내용은 f3의 내용과 동일한 모습을 확인할 수 있다.**

mount



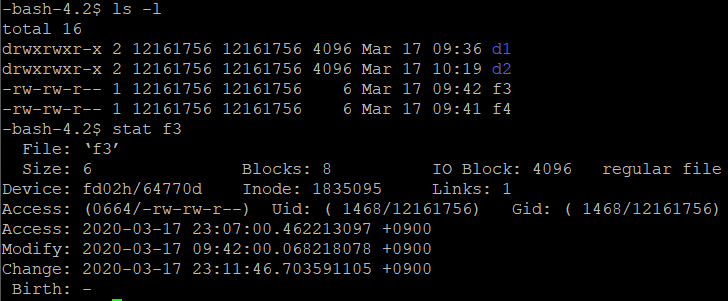
**$ mount : 현재의 마운트 목록을 출력한다.**

netstat



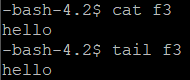
**$ netstat : 네트워킹과 연결된 시스템 정보를 출력한다.**

stat



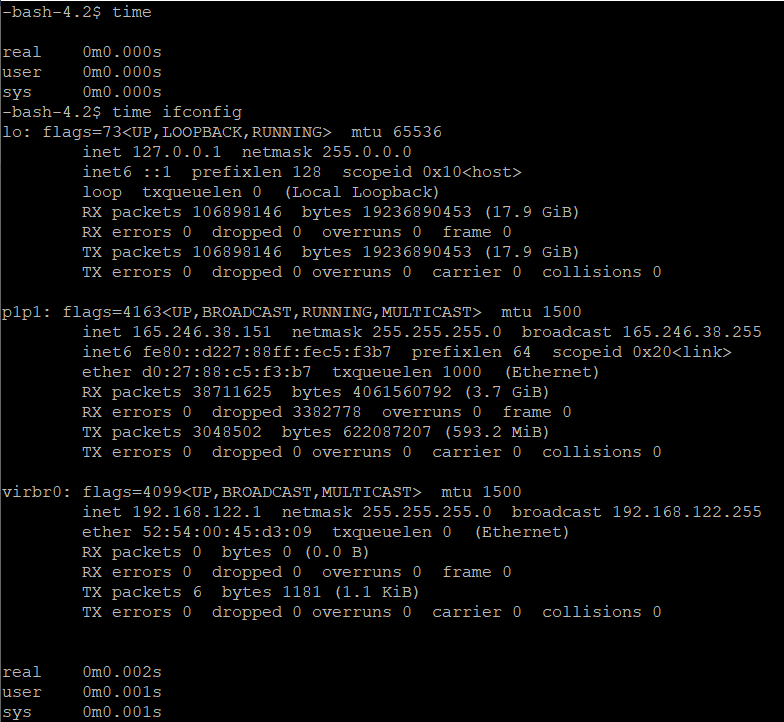
**$ stat : 파일, 파일시스템 상태를 출력한다.**

tail



**$ tail : [head]와는 반대로 아래에서 10줄(기본값) 내용을 출력한다.**

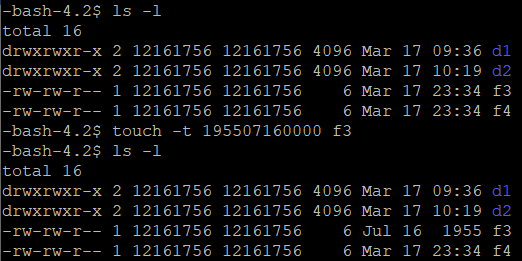
time



**$ time [명령어] : 명령어의 수행 시간을 측정한다.**

**= 캡처에서는 ifconfig의 명령어 수행 시간을 측정하였다.**

touch



**$ touch [옵션] 파일명 : 빈 파일을 생성하거나 파일의 날짜와 시간을 변경한다.**

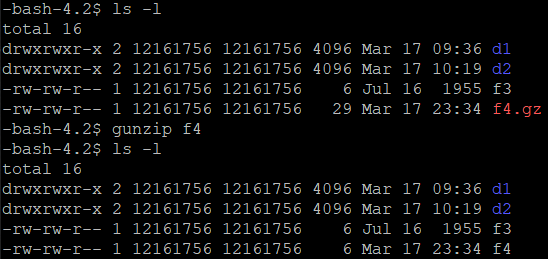
**= f3의 날짜와 시간이 1955년 07월 16일 00:00으로 변경된 것을 확인할 수 있다.**

tty



**$ tty : 현재 사용 중인 터미널 이름을 출력한다.**

gunzip



**$ gunzip [압축된 파일명] : 압축을 해제한다.**

**= f4.gz이 압축 해제되어 f4이 된 모습을 확인할 수 있다.**

whereis



**$ whereis [명령어] : 입력한 명령어의 $PATH 환경 변수에 설정된 실행 파일 경로와 메뉴얼 페이지의 전체 경로를 찾아준다.**

which



**$ which [명령어] 입력한 명령어의 $PATH 내의 실행 파일 위치를 알려준다.**

**= 명령어 [ls]는 alias 설정이 되어있고 [/usr/bin/ls]에 위치하는 것을 알 수 있다.**

whoami,



**$ whoami : 현재 로그인한 사용자 ID를 출력한다.**