

7/2

기본 명령2



cntl + z 는 멈춘 것 백그라운드에 대기 중

less,more : 긴 문서파일 보기

less 파일명

less는 뒤로 가기 가능

v 키를 입력하면 편집 가능

cat: 짧은 문서파일 보기, 짧은 문서 작성

cat 파일명 : 문서 보기

cat > 파일명 : 화면 입력하는 내용을 파일에 저장. 입력 후 cntl + d를 입력해서 저장하고 종료

>> : 기존 파일 내용에 추가

표준 파일 입력(0), 출력(1), 표준 에러(화면), redirection(표준 입출력, 에러의 방향을 바꿈)

```
autumn
winter[root@rocky ~]# cat test2
spring
summer
autumn
winter[root@rocky ~]# cat >> test2
melon
fruit[root@rocky ~]# cat test2
spring
summer
autumn
wintermelon
fruit[root@rocky ~]# cat > test3 < test2
[root@rocky ~]# cat test3
spring
summer
autumn
wintermelon
fruit[root@rocky ~]# |
```

>: 표준 출력 <: 표준 입력

```
[root@rocky ~]# a 2> error.txt
[root@rocky ~]# cat error.txt
-bash: a: command not found
[root@rocky ~]#
```

소유권(Ownership), 허가권(Permission)

소유권 : 파일이나 디렉토리의 소유자, 그룹을 표시, 소유자도 그룹도 아닌 사용자 그 외(Other) 해당 사용자가 파일에 접근하고자 할 때 소유권에 따라 접근 권한이 달라짐.

소유권을 확인하는 명령 : ls -l

```
leejeuk@DESKTOP-KM62UF8:~/ssh$ ls -l
total 32
-rw-r--r-- 1 leejeuk leejeuk 405 Jul 1 09:47 authorized_keys
-rw----- 1 leejeuk leejeuk 1831 Jul 1 09:46 id_rsa
-rw-r--r-- 1 leejeuk leejeuk 405 Jul 1 09:46 id_rsa.pub
-rw----- 1 leejeuk leejeuk 1831 Jul 1 10:38 id_rsa3
-rw-r--r-- 1 leejeuk leejeuk 405 Jul 1 10:38 id_rsa3.pub
-rw----- 1 leejeuk leejeuk 3076 Jul 1 14:22 known_hosts
-rw----- 1 leejeuk leejeuk 2240 Jul 1 14:22 known_hosts.old
-r----- 1 leejeuk leejeuk 1678 Jul 1 10:41 ljk-keypair.pem
```

3개씩 유저 - 그룹 - 그 외

소유자(USER), 그룹(GROUP), 그 외(OTHER) 별로 권한을 가짐.

허가권(Permission): 파일이나 디렉토리에 대한 권한(Read, Write, eXecute)

파일 : r(읽기), w(쓰기), x(실행)

디렉토리 : r(목록 읽기), w(디렉토리에 파일 쓰기), x(cd로 접근하기)

4 2 1 이다.

허가권은 user, group, other 별로 부여됨.

rwX rwX 퍼미션을 숫자로 표시 7 7 7

user group other

예) 644 = rw- r- -r- -

소유권 변경 : chown new_user.new_group 파일명

chgrp new_group 파일명

허가권 변경 : chmod 숫자 방식

문자 방식

숫자 방식은 모든 사용자별로 모두 작성 : 755, 640, 400

문자 방식은 변경되는 사용자만 작성 : u+x, o-w, ug +x, a=rw

+: 기존꺼에 더함

-: 기존꺼에 뺀다

=: 기존꺼 무시하고 부여

a: all

사용자를 특정 그룹에 포함시킬 경우

-a: add

G: 서브 그룹

usermod -aG 그룹명 사용자 : 이 사용자를 이 그룹을 2차 그룹으로 하게끔 가입시키겠다.

특수 퍼미션: setuid, setgid, stickybit

setuid, setgid : 실행 파일을 실행할 때 실행 파일의 소유자 권한이 아닌 실행하는 사용자의 권한으로 실행이 되는 것이 일반적인 상황.

소유자 혹은 소유 그룹의 권한으로 실행하는 것이 필요할 경우 setuid, setgid 퍼미션을 부여해야함.

stickybit: 디렉토리에 부여하는 특수퍼미션으로 해당 퍼미션이 부여된 디렉토리에는 모든 사용자가 쓰기 가능하지만 삭제는 소유자만 가능

4755(setuid), 2755(setgid), 1755(stickybit) : 특수 퍼미션이 붙으면 숫자가 4개이다 앞에 붙는다.

실행권한이 잘못 걸리면 대문자 S로 나옴

```
admin@rocky ~]$ [admin@rocky ~]$ sudo chmod 4755 vim
admin@rocky ~]$ ls -l
합계 2996
-rwsr-xr-x. 1 root root 3067776 7월 1 07:41 vim
admin@rocky ~]$ ./vim /etc/shadow
admin@rocky ~]$ ls -l /tmp
합계 8
-rwx-----. 1 root root 701 7월 1 07:43 ks-script-l
-rwx-----. 1 root root 291 7월 1 07:43 ks-script-n
-rwx-----. 3 root root 17 7월 1 2024 systemd-pri
admin@rocky ~]$ ls -ld /tmp
rwxrwxrwt. 8 root root 4096 7월 1 07:15 /tmp
admin@rocky ~]$ sudo chmod 4644 vim
admin@rocky ~]$ ls -l
합계 2996
-rwsr--r--. 1 root root 3067776 7월 1 07:41 vim
```

프로세스와 서비스

프로세스: 프로그램이 실행되어 메인 메모리에서 동작하고 있는 상태

서비스 : 데몬이라고도 하며 백그라운드에서 동작하면서 요청이 있을 경우 응답하는 프로세스

ps : 현재 터미널에서 동작시킨 프로세스 확인

ps -ef: 현재 시스템에서 동작 중인 모든 프로세스

ps aux: 메모리 cpu 점유율 확인

kill -l : 모든 시그널 출력

& : 백그라운드 동작

jobs를 하면 현재 실행중인 작업 보여

```
25188 pts/0 00:00:00 ps
[admin@rocky tmp]$ jobs
[1]+  Stopped                  sleep 1000
[2]-  Running                  sleep 2000 &
[admin@rocky tmp]$
```

fg % 작업 번호 : 백그라운드 작업 포그라운드로

bg % 작업 번호: 포그라운드 작업 백그라운드로

kill % 작업번호 : 작업을 제거한다.

```
[admin@rocky tmp]$ jobs
[2]+  Running                  sleep 2000 &
[admin@rocky tmp]$ kill %2
[admin@rocky tmp]$ jobs
[2]+  Terminated              sleep 2000
[admin@rocky tmp]$
```

kill -9 % 작업번호 : 작업을 죽인다.

```
[admin@rocky tmp]$ sleep 1000 &
[1] 25188
[admin@rocky tmp]$ jobs
[1]+  Running                  sleep 1000 &
[admin@rocky tmp]$ kill -9 %1
[admin@rocky tmp]$ jobs
[1]+  Killed                    sleep 1000
[admin@rocky tmp]$
```

SIGHUP(1) : 프로세스 일시중지했다가 다시 실행. 설정파일을 다시 읽어들임.

그냥 다시 꺾다 키는 것과 달리 프로세스 번호가 변하지 않는다.

SIGINT(2) : Ctrl + C에 의해 발생. 강제 종료

SIGKILL(9) : 강제 종료

SIGTERM(15) : 기본 시그널. 정상 종료

SIGTSTP(20): 일시 정지

sudo kill 시그널 번호 프로세스 번호

서비스(데몬) : 서비스를 제어하는 명령 - systemctl, service

systemctl 명령 서비스명.service ⇒ .service 생략 가능

- .socket
- start : 서비스 시작
- stop : 서비스 종료
- status : 서비스 상태 확인
- restart : 서비스 재시작
- enable : 부팅시 서비스 시작
- disable : 부팅시 서비스 시작 안함

root가 아니면 sudo는 붙여야

예) systemctl start docker = service docker start

systemctl start telnet.socket = service telnet.socket start

Rocky Linux는 SELinux 가 활성화 돼 있음. 비활성화하고 사용할 경우 설정

vi /etc/sysconfig/selinux

SELINUX = enforcing ⇒ disabled 하고 저장한 후 리부팅 영구적으로 적용

setenforce 0 ⇒ 일시적으로 해제

job control

현재 터미널에서 동작중인 작업 제어

jobs : 작업 목록 출력

fg : 중단된 프로세스(작업)을 포그라운드에서 동작

fg에서 동작인 프로세스 중단시킬 때 cntl + z

fg %작업번호 : 현재 멈춰있는 작업을 포그라운드로 동작
% 생략가능

bg : 중단된 프로세스 (작업)을 백그라운드에서 동작

bg %작업번호 : 현재 멈춰있는 작업을 백그라운드로 동작

명령 & : 해당 명령을 백그라운드에서 동작시킴.

kill %작업번호 : 해당 작업 종료

작업 자동화 : at, cron

at: 특정 시점에 수행할 작업을 지정하고 해당 시점에서 자동으로 작업 수행

cron: 주기적으로 수행할 작업을 지정하고 매 시점에서 해당 작업 수행

at 날짜,시간 지정

at 14:17 today

실행할 작업1

실행할 작업2

cnt

- 서버 시간을 시간 서버와 동기화 할 경우 :sudo systemctl restart chronyd
- timezone 변경: timedatectl set-timezone Asia/Seoul

crontab -e : 반복적으로 수행할 작업 등록(vi)

분 시간 일 월 요일(0~6)

10 * * * * 수행할 명령 등록

*/20 * * * * 수행할 쉘스크립트 파일 등록

at, cron에 대한 접근 제어 :

at.deny , at.allow, cron.deny, cron.allow

.deny : 해당 파일에 등록된 사용자는 at, cron 사용 불가

.allow : 해당 파일에 등록된 사용자만 at,cron 사용 가능

.deny, .allow 파일이 모두 있는 경우 .allow 가 적용 됨

내용이 비어 있는 경우 모두 허용

sudo

일반 사용자가 root 권한으로 명령을 수행할 필요가 있는 경우 /etc/sudoers 파일에 사용자를 등록하고 사용할 수 있는 명령을 지정함. su 명령으로 root로 사용자를 전환하고 명령을 수행할 수 있으나 특정 명령만 허용할 수 없음.

visudo : /etc/sudoers 파일을 수정할 수 있음

tar, 압축프로그램

tar(tape archive) : 여러 파일들을 묶어서 하나의 파일로 만드는 유틸리티

옵션

c : 파일 묶기

x : 묶인 파일 풀기

v : 묶거나 푸는 과정 출력

t : 실제로 풀거나 묶지 않고 내용 확인

f : 장치나 파일 지정

예) tar cvf test.tar c*

tar xvf test.tar

tar tvf test.tar

tar xvf test.tar -C /tmp -c 옵션은 tar 해제 위치 지

ex)

tar cvf cfile.tar c* c로 시작하는 파일 cfile.tar로 묶는다.

tar cf pfile.tar p* 묶되 과정 출력 x

압축프로그램 : gzip(gunzip), bzip(bunzip2), xz, compress, zip(unzip)

압축시 :

gzip 파일명

bzip2 파일명

xz 파일명

compress 파일명

zip 파일명.zip 압축할 파일 명

```
zip error: Zip file structure invalid (cfile5.tar)
[admin@rocky workdir]$ ls -lh cfile?.*
-rw-rw-r--. 1 admin admin 3.9M  7월  2 03:29 cfile1.tar.gz
-rw-rw-r--. 1 admin admin 3.6M  7월  2 03:29 cfile2.tar.bz2
-rw-rw-r--. 1 admin admin 768K  7월  2 03:29 cfile3.tar.xz
-rw-rw-r--. 1 admin admin 5.6M  7월  2 03:29 cfile4.tar.Z
-rw-rw-r--. 1 admin admin 9.6M  7월  2 03:30 cfile5.tar
[admin@rocky workdir]$
```

압축 결과 xz가 가장 압축률이 높음을 확인 가능

압축해제시 :

gunzip 파일명

gzip -d 파일명

bunzip2 파일명

bzip2 -d 파일명

xz -d 파일명

uncompress 파일명

unzip 파일명.zip 압축할 파일명

tar와 압축프로그램을 함께 사용하여 tar 묶고 압축하는 과정을 한번에 처리

tar xvfz 파일명.tar.gz ⇒ gzip으로 압축해제하고 tar 파일 풀기 동시에 진행.(z 생략가능)

tar xvfj 파일명.tar.bz2 ⇒ bzip2으로 압축해제하고 tar 파일 풀기 동시에 진행.

tar xvfJ 파일명.tar.xz ⇒ xz으로 압축해제하고 tar 파일 풀기 동시에 진행.

tar xvfZ 파일명.tar.Z ⇒ compress로 압축해제하고 tar 파일 풀기 동시에 진행

파일 시스템

물리적인 하드디스크에 파일 저장하기 위한 체계

리눅스 파일시스템 : ext2, ext3, ext4, xfs

파일시스템 확인 : df -Th

리눅스에서는 모든 장치를 파일로 취급함. 그리고 모든 장치는 마운트가 되어야 사용할 수 있음.

시스템 종료 명령: poweroff, halt, shutdown -h now , init 0

시스템 리부팅 명령: reboot, shutdown -r now, init 6

mount -t ext4 /dev/sda1 /mnt/mpoint/

/mnt/mpoint에 /dev/sda1을 연결 연결할 때 파일 시스템은 ext4이다.

앞으로 이 장치에 연결하고 싶으면 /mnt/mpoint을 통해 연결 가능하다.

새로운 하드 디스크를 추가한 경우

1) 파티션 생성

파티션은 파일 시스템을 생성하기 위한 구역

fdisk 장치파일 : fdisk /dev/sda

2) 파일 시스템

파일 시스템은 파일을 저장하기 위한 체계. os마다 다름.

mkfs -t 파일시스템종류 파티션 : mkfs -t ext4 /dev/sda1

3) 생성된 파일 시스템을 리눅스에서 사용 가능하도록 mount함

mount는 장치를 리눅스에서 사용 가능하도록 연결하는 것

mount 연결할 파티션 연결지점 : mount /dev/sda1 /mnt/mpoint

umount 연결지점 또는 연결된 파티션 : umount /mnt/mpoint

4) 시스템 리부팅 후에도 지속적으로 mount 상태로 사용하기 위해 /etc/fstab 파일에 등록해야 함.

파티션 또는 장치	연결지점	파일시스템	옵션	dump여부	fsck순서
/dev/sda1	/mnt/mpoint	ext4	defaults	0	0

fsck : 파일 시스템 체크 0은 안하겠다. 1~순서 체크해서 해당 하면 검사

```
xfce4-appfinder xfce4-terminal xfce4-notify xfce4-mouseemulation xfce4-mouse xfce4-keyboard xfce4-xfce4-panel
[root@rocky ~]# fdisk /dev/sda

Welcome to fdisk (util-linux 2.32.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): p
Disk /dev/sda: 10 GiB, 10737418240 bytes, 20971520 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x60b919c0

Device     Boot Start      End  Sectors  Size Id Type
/dev/sda1  2048 20971519 20969472  10G 83 Linux

Command (m for help): d
Selected partition 1
Partition 1 has been deleted.

Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): |
```

-p 현재 있는 디스크 보여주기

-n 새로 만들기

-d 지우기

-w 저장하고 나오기

primary는 최대 4개까지 더 하고 싶으면 extended

RAID : 하드디스크를 묶어서 하나로 사용

묶어진 것을 VOLUME이라 한다.

리눅스에서는 LVM을 사용한다.

장점

용량 확장, 장애 대비(백업)

파티션을 만들고 설정이 다 되어있는 상태에서 vi /etc/fstab으로 들어가서 수정

```
fdisk /dev/sda
fdisk -l /dev/sda
mkfs -t ext4 /dev/sda1
mkfs -t xfs /dev/sda3

mount -t ext4 /dev/sda1 /mnt/mpoint1
mount -t xfs /dev/sda3 /mnt/mpoint2
```

```
#
# /etc/fstab
# Created by anaconda on Mon Jul 1 11:31:04 2024
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rl-root    /                    xfs      defaults    0 0
UUID=8e5324f9-8594-4606-8937-dbd619752c93 /boot               xfs      defaults    0 0
/dev/mapper/rl-swap    none                swap      defaults    0 0
/dev/sda1              /mnt/mpoint1        ext4     defaults    0 0
/dev/sda3              /mnt/mpoint3        xfs      defaults    0 0
```



df -Th로 현재 디스크들 확인 가능하다.

```
[root@rocky ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
devtmpfs	devtmpfs	1.8G	0	1.8G	0%	/dev
tmpfs	tmpfs	1.8G	0	1.8G	0%	/dev/shm
tmpfs	tmpfs	1.8G	8.9M	1.8G	1%	/run
tmpfs	tmpfs	1.8G	0	1.8G	0%	/sys/fs/cgroup
/dev/mapper/rl-root	xfs	26G	3.5G	23G	14%	/
/dev/nvme0n1p1	xfs	1014M	199M	816M	20%	/boot
tmpfs	tmpfs	364M	0	364M	0%	/run/user/0
/dev/sda1	ext4	4.9G	24K	4.6G	1%	/mnt/mpoint1
/dev/sda3	xfs	5.0G	68M	5.0G	2%	/mnt/mpoint2