

# 리눅스

---

발표자 : 201232038 이주형  
201332012 김재현  
201532038 정현진  
201532056 박다솔  
201532005 김도희

# 15장. 리눅스 파일시스템관리

## 파일시스템 포맷하기

ext4(extended file system) 타입으로 파일시스템 포맷하기

```
mke2fs -t ext4 /dev/sdb1
```

ext3(extended file system) 타입으로 파일시스템 포맷하기

```
mke2fs -t ext3 /dev/sdb1
```

옵션 -t를 입력하지 않으면 기본적으로 ext2타입

옵션 -b: 블록크기를 지정, 기본 4096bytes

옵션 -c: 배드블록을 점검후 파일시스템 생성

옵션 -f: 프래그먼트 크기를 지정

# 15장. 리눅스 파일시스템관리

## 마운트

마운트란?

리눅스 시스템에서 사용하기를 원하는  
특정장치(디바이스)를 시스템에 인식시키는 작업  
디렉토리와 연결하는 작업  
특정장치를 하나의 디렉토리처럼 사용하기 위한 작업

# 15장. 리눅스 파일시스템관리

## 마운트 설정파일

```
[root@host ~]# mount
/dev/sda2 on / type ext4 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
/dev/sda1 on /boot type ext4 (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
vmware-vmblock on /var/run/vmblock-fuse type fuse.vmware-vmblock (rw,nosuid,nodev,default_permissions,allow_other)
```

```
[root@host ~]# cat /etc/mtab
/dev/sda2 / ext4 rw 0 0
proc /proc proc rw 0 0
sysfs /sys sysfs rw 0 0
devpts /dev/pts devpts rw,gid=5,mode=620 0 0
tmpfs /dev/shm tmpfs rw,rootcontext="system_u:object_r:tmpfs_t:s0" 0 0
/dev/sda1 /boot ext4 rw 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
vmware-vmblock /var/run/vmblock-fuse fuse.vmware-vmblock rw,nosuid,nodev,default_permissions,allow_other 0 0
```

# 15장. 리눅스 파일시스템관리

## 마운트

국제 표준화 기구(ISO)에서 제정한 CD-ROM 매체를 위한 파일 시스템 표준이다.

### CD-ROM 마운트

```
mount -t iso9660 /dev/cdrom /media/cdrom
```

### CD-ROM 언마운트

```
umount /media/cdrom
```

마운트 상태에서 CD-ROM을 뺄수 없음  
/media/cdrom에서 다른곳으로 빠져 나와서 umount 해야한다.

# 15장. 리눅스 파일시스템관리

## 마운트

옵션	설명
-r	특정장치를 마운트 할때 읽을수만 있도록 마운트(쓰기 금지)(read only)
-w	읽기,쓰기 모두 가능하도록 마운트

# 15장. 리눅스 파일시스템관리

어떤 디바이스를 부팅과정에서 마운트 시킬것인가?

```
[root:~, 2]# cat /etc/fstab
```

```
#
# /etc/fstab
# Created by anaconda on Tue Oct  4 18:11:10 2016
#
# 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
#
dev/mapper/Vo1Group-lv_root / ext4 defaults 1 1
UUID=23bb8c47-14da-4a43-8b7f-e00c2f0b183d /boot ext4 defaults 1 2
/dev/mapper/Vo1Group-lv_home /home ext4 defaults 1 2
/dev/mapper/Vo1Group-lv_swap swap swap defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
```

마운트포인트

시스템타입

디바이스명

# 15장. 리눅스 파일시스템관리

e2fsck

e2fsck: 파일시스템 점검과 복구

<점검 항목>

Inode

Blocks

Sizes

디렉토리구조

디렉토리 연결성

파일 링크정보

전체파일 개수

전체 블록수중 사용중인 블록



# 15장. 리눅스 파일시스템관리

## 파일 시스템 점검, 복구

### 파일시스템 강제 점검 및 복구하기

```
e2fsck -f /dev/hdc2
```

```
[root@host ~]# e2fsck /dev/sda1  
e2fsck 1.41.12 (17-May-2010)  
/dev/sda1 is mounted.
```

```
WARNING!!! The filesystem is mounted. If you continue you ***WILL***  
cause ***SEVERE*** filesystem damage.
```

```
Do you really want to continue (y/n)? yes
```

```
/dev/sda1: recovering journal  
/dev/sda1: clean, 39/76912 files, 39729/307200 blocks
```

# 15장. 리눅스 파일시스템관리

## 디스크 사용량 점검 df

```
[root@localhost ~]# df -ah
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2        47G   2.2G   43G   5% /
proc            0      0      0  -  /proc
sysfs            0      0      0  -  /sys
devpts          0      0      0  -  /dev/pts
tmpfs           504M    88K   504M   1% /dev/shm
/dev/sda1       291M    30M   247M  11% /boot
none             0      0      0  -  /proc/sys/fs/binfmt_misc
vmware-vmblock  0      0      0  -  /var/run/vmblock-fuse
```

-h: 사람이 보기 쉽도록 최적의 용량단위로 표시  
(human-readable)

# 15장. 리눅스 파일시스템관리

디렉토리별 사용량 점검 du

```
[root@localhost etc]# du -sh /etc  
32M      /etc
```

-s: 지정된 디렉토리내에 존재하는 모든파일과  
서브디렉토리들의 용량을 모두 합친 용량  
(summary)

# 15장. 리눅스 파일시스템관리

## 디렉토리별 사용량 점검 du

```
[root@localhost ~]# du -sh /tmp/
324K    /tmp/
[root@localhost ~]# du -sh /tmp/*
4.0K    /tmp/ks-script-dvkd2x
4.0K    /tmp/ks-script-dvkd2x.log
8.0K    /tmp/orbit-gdm
8.0K    /tmp/pulse-s471oKtjECIJ
8.0K    /tmp/pulse-u9usfHfI33FW
4.0K    /tmp/virtual-lee.LMTGYe
8.0K    /tmp/vmware-config0
4.0K    /tmp/VMwareDnD
12K     /tmp/vmware-lee
48K     /tmp/vmware-root
196K    /tmp/vmware-root-2794398616
0       /tmp/yum.log
```

# 15장. 파일시스템 관리 실무

/etc/fstab

## ✓ /etc/fstab 파일의 구성

➤ [파일시스템장치명] [마운트포인트] [파일시스템종류]  
[옵션] [dump 관련설정] [파일점검옵션]

```
[root:~, 9]# cat /etc/fstab
```

```
#
# /etc/fstab
# Created by anaconda on Tue Oct  4 18:11:10 2016
#
# Accessible filesystems, by reference, are maintained under '/dev/di
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for mor
#
/dev/mapper/VolGroup-lv_root /                ext4      defaults
UUID=23bb8c47-14da-4a43-8b7f-e00c2f0b183d /boot    ext4      defaults
/dev/mapper/VolGroup-lv_home /home      ext4      defaults
/dev/mapper/VolGroup-lv_swap swap        swap       defaults
tmpfs                /dev/shm   tmpfs      defaults
devpts               /dev/pts   devpts     gid=5,mode=62
sysfs                /sys       sysfs      defaults
proc                 /proc      proc       defaults
[root:~, 10]#
```



```
[root@host ~]# cat /proc/filesystems
nodev    sysfs
nodev    rootfs
nodev    bdev
nodev    proc
nodev    cgroup
nodev    cpuset
nodev    tmpfs
nodev    devtmpfs
nodev    binfmt_misc
nodev    debugfs
nodev    securityfs
nodev    sockfs
nodev    usbfs
nodev    pipefs
nodev    anon_inodefs
nodev    inotifyfs
nodev    devpts
nodev    ramfs
nodev    hugetlbfs
nodev    iso9660
nodev    mqueue
nodev    selinuxfs
nodev    ext4
nodev    vmhgfs
nodev    fuse
nodev    fuseblk
nodev    fusectl
[root@host ~]#
```

# 15장. 파일시스템 관리 실무

## /etc/fstab

```
[root:~, 9]# cat /etc/fstab
```

```
#
# /etc/fstab
# Created by anaconda on Tue Oct  4 18:11:10 2016
#
# 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
#
/dev/mapper/VolGroup-lv_root / ext4 defaults 1 1
UUID=23bb1c47-14da-4a43-8b7f-e00c2f0b183d /boot ext4 defaults 1 2
/dev/mapper/VolGroup-lv_home /home ext4 defaults 0 0
/dev/mapper/VolGroup-lv_swap swap swap defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
[root:~, 10]#
```

✓ [파일점검옵션] : /etc/fstab의 여섯번째 항목

- 0 : fsck가 실행되지 않음
- 1 : 루트파일시스템
- 2 : 루트파일시스템 이외

# 15장. 파일시스템 관리 실무

## 스왑메모리

### ✓ 스왑메모리란?

➤ 리눅스에서의 가상메모리

➤ 관련명령어

I. mkswap : 리눅스의 스왑영역 지정(스왑공간 생성)

II. swapon : 스왑공간 활성화

III. sawpoff : 스왑공간 비활성화

➤ 스왑공간 추가

I. 스왑파티션을 스왑공간으로 추가

II. 스왑파일 생성 후, 스왑공간으로 추가

# 15장. 파일시스템 관리 실무

## 스왑메모리

### 1. 현재 시스템 메모리 상태 확인

```
[root@localhost ~]# free
```

	total	used	free
Mem:	1031204	362536	668668
-/+ buffers/cache:		156736	874468
Swap:	2064376	0	2064376

스왑메모리:  
2064376 바  
이트

### 2. 스왑파티션 생성

```
[root@localhost ~]# fdisk /dev/sda
```

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to switch off the mode (command 'c') and change display units to sectors (command 'u').

Command (m for help): **p**

Disk /dev/sda: 53.7 GB, 53687091200 bytes  
255 heads, 63 sectors/track, 6527 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x0003f428

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	39	307200	83	Linux
Partition 1 does not end on cylinder boundary.						
/dev/sda2		39	6271	50056192	83	Linux
/dev/sda3		6271	6528	2064384	82	Linux swap / Solaris

Command (m for help): **t**  
Partition number (1-4): **1**  
Hex code (type L to list codes): **82**  
Changed system type of partition 1 to 82 (Linux swap / Solaris)

Command (m for help): **p**

Disk /dev/sda: 53.7 GB, 53687091200 bytes  
255 heads, 63 sectors/track, 6527 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x0003f428

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	39	307200	82	Linux swap / Solaris
Partition 1 does not end on cylinder boundary.						
/dev/sda2		39	6271	50056192	83	Linux
/dev/sda3		6271	6528	2064384	82	Linux swap / Solaris

Command (m for help): **w**  
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource busy.  
The kernel still uses the old table. The new table will be used at the next reboot or after you run partprobe(8) or kpartx(8)  
Syncing disks.  
[root@localhost ~]#



# 15장. 파일시스템 관리 실무

## 스왑메모리

### 3. 스왑파티션을 스왑영역으로 생성하는 포맷작업

```
[root@host ~]# mkswap -c -v1 /dev/sda3
Setting up swapspace version 1, size = 2064380 KiB
no label, UUID=b6083e5d-11b6-4509-bef1-212dfef473a4
```

### 4. 스왑영역 활성화하기

```
[root@host ~]# swapon /dev/sda3
```

### 5. 파일시스템테이블 파일에 등록하는 작업

```
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.
```

```
touch /var/lock/subsys/local
swapon /dev/sda3
```

tmpfs	/dev/shm	tmpfs	defaults	0 0
devpts	/dev/pts	devpts	gid=5,mode=620	0 0
sysfs	/sys	sysfs	defaults	0 0
proc	/proc	proc	defaults	0 0
/dev/sda3	swap	swap	swap	0 0

etc/rc.d/rc.local

# 15장. 파일시스템 관리 실무

## dumpe2fs

✓ 특정 파일시스템의 상세정보와 사용내역

### ➤ dumpe2fs

```
Journal backup:           inode blocks
Journal features:         (none)
Journal size:             8M
Journal length:           8192
Journal sequence:         0x0000001c
Journal start:            1
```

```
Group 0: (Blocks 1-8192) [ITABLE_ZEROED]
Checksum 0x6cf3, unused inodes 2007
Primary superblock at 1, Group descriptors at 2-3
Reserved GDT blocks at 4-259
Block bitmap at 260 (+259), Inode bitmap at 276 (+275)
Inode table at 292-544 (+291)
3836 free blocks, 2007 free inodes, 2 directories, 2007 unused inodes
Free blocks: 4357-8192
Free inodes: 18-2024
Group 1: (Blocks 8193-16384) [INODE_UNINIT, ITABLE_ZEROED]
```

# 15장. 파일시스템 관리 실무

## dumpe2fs

### ✓ 파일시스템의 슈퍼블록

#### ➤ dumpe2fs -h

```
[root@localhost ~]# dumpe2fs /dev/sda1 | grep superblock
```

```
dumpe2fs 1.41.12 (17-May-2010)
```

```
Primary superblock at 1, Group descriptors at 2-3
```

```
Backup superblock at 8193, Group descriptors at 8194-8195
```

```
Backup superblock at 24577, Group descriptors at 24578-24579
```

```
Backup superblock at 40961, Group descriptors at 40962-40963
```

```
Backup superblock at 57345, Group descriptors at 57346-57347
```

```
Backup superblock at 73729, Group descriptors at 73730-73731
```

```
Backup superblock at 204801, Group descriptors at 204802-204803
```

```
Backup superblock at 221185, Group descriptors at 221186-221187
```

```
[root@localhost ~]#
```

```
Reserved GDT blocks: 256
```

```
Blocks per group: 8192
```

```
Fragments per group: 8192
```

```
Inodes per group: 2024
```

```
Inode blocks per group: 253
```

```
Flex block group size: 16
```

```
Filesystem created: Wed Jan 18 05:41:56 2017
```

```
Last mount time: Sun Jan 22 20:32:46 2017
```

```
Last write time: Sun Jan 22 20:32:46 2017
```

```
Mount count: 5
```

# 15장. 파일시스템 관리 실무

## tune2fs

### ✓ 파일시스템 튜닝명령어(tune2fs)

#### ➤ 파일 시스템의 슈퍼블록 정보 확인

```
[root@localhost ~]# tune2fs -l /dev/sda1
tune2fs 1.41.12 (17-May-2010)
Filesystem volume name:   <none>
Last mounted on:         /boot
Filesystem UUID:         c09c792a-b0de-4e70-9242-39a5bcdafc0a
Filesystem magic number:  0xEF53
Filesystem revision #:    1 (dynamic)
Filesystem features:      has_journal ext_attr resize_inode dir
_bg sparse_super huge_file uninit_bg dir_nlink extra_isize
Filesystem flags:         signed_directory_hash
Default mount options:    user_xattr acl
Filesystem state:         clean
Errors behavior:          Continue
Filesystem OS type:       Linux
Inode count:              76912
Block count:              307200
```

# 15장. 파일시스템 관리 실무

## tune2fs

### ✓ 파일시스템의 최대 마운트 횟수 지정

```
[root@localhost ~]# tune2fs -c 828 /dev/sda1
tune2fs 1.41.12 (17-May-2010)
Setting maximal mount count to 828
[root@localhost ~]#
```

```
[root@localhost ~]# tune2fs -l /dev/sda1 | grep mount
Last mounted on:          /boot
Default mount options:    user_xattr acl
Last mount time:          Sun Jan 22 20:32:46 2017
Maximum mount count:      828
[root@localhost ~]#
```

### ✓ 예비블록 개수 & 퍼센트 설정

```
[root@localhost ~]# tune2fs -r 25000 /dev/sda1
tune2fs 1.41.12 (17-May-2010)
Setting reserved blocks count to 25000
[root@localhost ~]#
```

```
[root@localhost ~]# tune2fs -m 28 /dev/sda1
tune2fs 1.41.12 (17-May-2010)
Setting reserved blocks percentage to 28% (86016 blocks)
[root@localhost ~]#
```

# 15장. 파일시스템 관리 실무

## tune2fs

### ✓ 예비블록 사용자 & 그룹설정

```
[root@localhost ~]# tune2fs -l /dev/sda1 | grep Reserved
Reserved block count:      86016
Reserved GDT blocks:      256
Reserved blocks uid:       0 (user root)
Reserved blocks gid:       0 (group root)
[root@localhost ~]# tune2fs -u lee /dev/sda1
tune2fs 1.41.12 (17-May-2010)
Setting reserved blocks uid to 500
[root@localhost ~]# tune2fs -l /dev/sda1 | grep Reserved
Reserved block count:      86016
Reserved GDT blocks:      256
Reserved blocks uid:       500 (user lee)
Reserved blocks gid:       0 (group root)
[root@localhost ~]# tune2fs -g root /dev/sda1
tune2fs 1.41.12 (17-May-2010)
Setting reserved blocks gid to 0
[root@localhost ~]# tune2fs -l /dev/sda1 | grep Reserved
Reserved block count:      86016
Reserved GDT blocks:      256
Reserved blocks uid:       500 (user lee)
Reserved blocks gid:       0 (group root)
[root@localhost ~]#
```

# 15장. 파일시스템 관리 실무

debugfs

## ✓ 파일시스템 디버거(debugfs)

- debugfs는 root 전용명령어
- 슈퍼블럭과 inode를 조작
- 장치명은 실제 장치명으로
- ext2 ext3로 생성된 파일시스템만을 대상으로 디버깅하거나 삭제파일 복구

➤ **native filesystem 모드** : 셸프롬프트가 존재하는 쉘상태

-> 전용 명령어 : 1cd

➤ **debugfs 모드** -> 전용 명령어 : cd

```
[root@localhost ~]# debugfs
debugfs 1.41.12 (17-May-2010)
debugfs:
debugfs: q
[root@localhost ~]#
```

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ debugfs 전용모드에서 파일시스템 열고 닫기

```
[root@localhost ~]# debugfs
debugfs 1.41.12 (17-May-2010)
debugfs:
debugfs: open -w /dev/sda1
debugfs:
debugfs: ls
 2 (12) .      2 (12) ..     11 (20) lost+found  32385 (12) grub
32387 (56) efi   12 (48) .vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
debugfs: 
debugfs: close filesystems
debugfs:
debugfs: ls
ls: Filesystem not open
debugfs: 
```



# 15장. 파일시스템 관리 실무

## debugfs

- ✓ debugfs 읽기 모드와 쓰기모드
- 옵션 : -w의 유무에 따라 읽기모드와 쓰기모드로 나뉜다.

## ✓ 현재 작업 디렉토리 출력

```
[root@localhost ~]# debugfs -w /dev/sda1
debugfs 1.41.12 (17-May-2010)
debugfs:
debugfs: pwd
[pwd]      INODE:      2  PATH: /
[root]     INODE:      2  PATH: /
debugfs:
debugfs: print_working_directory
[pwd]      INODE:      2  PATH: /
[root]     INODE:      2  PATH: /
debugfs: ■
```

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 디렉토리 리스트 확인

```
debugfs ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (56) efi 12 (48) .vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
debugfs:
debugfs: q
```

```
[root@localhost ~]# ls -l /boot
total 19283
-rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
-rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
drwx-----. 2 root root 12288 Jan 18 05:41 lost+found
drwxr-xr-x. 2 root root 1024 Jan 22 22:33 newdir
-rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
-rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
-rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
[root@localhost ~]#
```

### ✓ 파라미터 보기

```
[root@localhost ~]# debugfs
debugfs 1.41.12 (17-May-2010)
debugfs:
debugfs: params
Filesystem in use: --none--
debugfs: 
```

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 새 디렉토리 만들기

```
[root@localhost ~]# debugfs -w /dev/sda1
debugfs 1.41.12 (17-May-2010)
debugfs:
debugfs: mkdir newdir1
debugfs:
debugfs: q
[root@localhost ~]# ls -l /boot
total 19284
-rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
-rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
drwx-----. 2 root root 12288 Jan 18 05:41 lost+found
drwxr-xr-x. 2 root root 1024 Jan 22 22:33 newdir
drwxr-xr-x. 2 root root 1024 Jan 22 22:38 newdir1
-rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
-rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
-rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
[root@localhost ~]#
```

### ✓ 현재 작업디렉토리 변경

```
debugfs: ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (12) efi 18 (16) newdir 19 (28) newdir1
12 (48) .vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
debugfs: cd newdir
debugfs: pwd
[pwd] INODE: 18 PATH: /newdir
[root@localhost ~]# pwd
/pwd INODE: 2 PATH: /
debugfs:
```

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 실행 전 디렉토리로 이동

```
debugfs: lcd /tmp
debugfs:
debugfs: pwd
[pwd] INODE:      18  PATH: /newdir
[root] INODE:      2  PATH: /
debugfs:
debugfs: lcd /boot
debugfs:
debugfs: pwd
[pwd] INODE:      18  PATH: /newdir
[root] INODE:      2  PATH: /
debugfs:
debugfs: q
```

- native filesystem -> 셸프롬프트가 존재하는 디렉토리 경로
- "cd" -> debugfs에서의 현재작업위치 변경
- "lcd"-> "native filesystem"위치를 변경
- 따라서, 작업위치는 변하지 않았다.

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 특정 디렉토리 삭제

```
debugfs: ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (12) efi 18 (16) newdir 19 (28) newdir1
12 (48) .vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
```

debugfs:

debugfs: rmdir newdir1

debugfs:

```
debugfs: ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (12) efi 18 (44) newdir
```

[root@localhost ~]# ls -l /boot

total 19283

```
-rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
-rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
drwx----- 2 root root 12288 Jan 18 05:41 lost+found
drwxr-xr-x. 2 root root 1024 Jan 22 22:33 newdir
-rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
-rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
-rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
```

[root@localhost ~]#

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 파일복사하기

```
debugfs: write test test2
Allocated inode: 18
debugfs:
debugfs: ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (12) efi 18 (44) test2
12 (48) vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
debugfs: █
```

```
[root@localhost ~]# ls -l /boot
total 19283
-rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
-rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
drwx----- 2 root root 12288 Jan 18 05:41 lost+found
-rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
-rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 2 root root 1024 Jan 22 22:55 test2
-rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
[root@localhost ~]# █
```

### ✓ 삭제된 파일의 inode 목록 확인

```
debugfs: lsdel
Inode Owner Mode Size Blocks Time deleted
19 0 40755 1024 1/ 1 Sun Jan 22 22:48:22 2017
1 deleted inodes found.
debugfs: █
```

➤ inode 정보를 알고 있는 경우, 삭제된 파일을 복구할 수도 있다.

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 슈퍼블럭 정보 및 상태 확인

```
debugfs: stats
Filesystem volume name:    <none>
Last mounted on:          /boot
Filesystem UUID:          c09c792a-b0de-4e70-9242-39a5bcdafc0a
Filesystem magic number:  0xEF53
Filesystem revision #:    1 (dynamic)
Filesystem features:      has_journal ext_attr resize_inode dir_index fi
_bg sparse_super huge_file uninit_bg dir_nlink extra_isize
Filesystem flags:         signed_directory_hash
Default mount options:    user_xattr acl
Filesystem state:         clean
Errors behavior:          Continue
```

### ✓ 특정파일의 inode 상세정보 확인

```
debugfs: stat grub
Inode: 32385   Type: directory   Mode: 0755   Flags: 0x80000
Generation: 376332824   Version: 0x00000019
User:    0   Group:    0   Size: 1024
File ACL: 4354   Directory ACL: 0
Links: 2   Blockcount: 4
Fragment: Address: 0   Number: 0   Size: 0
ctime: 0x587ef5a3 -- Tue Jan 17 20:57:07 2017
atime: 0x5885a453 -- Sun Jan 22 22:36:03 2017
mtime: 0x587ef5a3 -- Tue Jan 17 20:57:07 2017
EXTENTS:
2021-01-09 13:51:53
debugfs: █
```

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 현재 사용가능한 debugfs 명령어

```
debugfs: lr
Available debugfs requests:

show_debugfs_params, params      Show debugfs parameters
open_filesys, open               Open a filesystem
close_filesys, close             Close the filesystem
feature, features                Set/print superblock features
dirty_filesys, dirty             Mark the filesystem as dirty
init_filesys                     Initialize a filesystem (DESTROYS DATA)
show_super_stats, stats          Show superblock statistics
ncheck                           Do inode->name translation
icheck                           Do block->inode translation
change_root_directory, chroot    Change root directory
change_working_directory, cd      Change working directory
list_directory, ls               List directory
show_inode_info, stat            Show inode information
dump_extents, extents, ex        Dump extents information
link, ln                         Create directory link
unlink                           Delete a directory link
mkdir                            Create a directory
rmdir                            Remove a directory
```

### ✓ 현재위치에서 가장 가까운 빈 블록 찾기

```
debugfs: ffb
Free blocks found: 4357
debugfs: █
```



# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 현재 위치에서 빈 inode 찾기

```
debugfs: ffi
Free inode found: 19
debugfs: █
```

### ✓ 파일링크 삭제하기

```
[root@localhost ~]# ls -l /boot
total 19282
-rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
-rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
drwx-----. 2 root root 12288 Jan 18 05:41 lost+found
-rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
-rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
-rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
[root@localhost ~]# █
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
debugfs: █
```

# 15장. 파일시스템 관리 실무

## debugfs

### ✓ 파일링크 생성하기

```
debugfs: ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (56) efi 12 (48) .vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
```

```
debugfs: link efi efi_new
```

```
debugfs:
debugfs: ls
2 (12) . 2 (12) .. 11 (20) lost+found 32385 (12) grub
32387 (12) efi 32387 (44) efi_new
12 (48) .vmlinuz-2.6.32-131.0.15.el6.i686.hmac
13 (44) System.map-2.6.32-131.0.15.el6.i686
14 (40) config-2.6.32-131.0.15.el6.i686
15 (44) symvers-2.6.32-131.0.15.el6.i686.gz
16 (40) vmlinuz-2.6.32-131.0.15.el6.i686
17 (696) initramfs-2.6.32-131.0.15.el6.i686.img
```

```
debugfs: □
[root@localhost boot]# ls -li /boot
total 19284
-rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi_new
drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
-rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
drwx-----. 2 root root 12288 Jan 18 05:41 lost+found
-rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
-rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
-rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
[root@localhost boot]# ls -li /boot
total 19284
14 -rw-r--r--. 1 root root 104961 Nov 12 2011 config-2.6.32-131.0.15.el6.i686
32387 drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi
32387 drwxr-xr-x. 3 root root 1024 Jan 18 05:50 efi_new
32385 drwxr-xr-x. 2 root root 1024 Jan 17 20:57 grub
17 -rw-r--r--. 1 root root 13910411 Jan 18 05:57 initramfs-2.6.32-131.0.15.el6.i686.img
11 drwx-----. 2 root root 12288 Jan 18 05:41 lost+found
15 -rw-r--r--. 1 root root 162891 Nov 12 2011 symvers-2.6.32-131.0.15.el6.i686.gz
13 -rw-r--r--. 1 root root 1796044 Nov 12 2011 System.map-2.6.32-131.0.15.el6.i686
16 -rwxr-xr-x. 1 root root 3745664 Nov 12 2011 vmlinuz-2.6.32-131.0.15.el6.i686
[root@localhost boot]# □
```

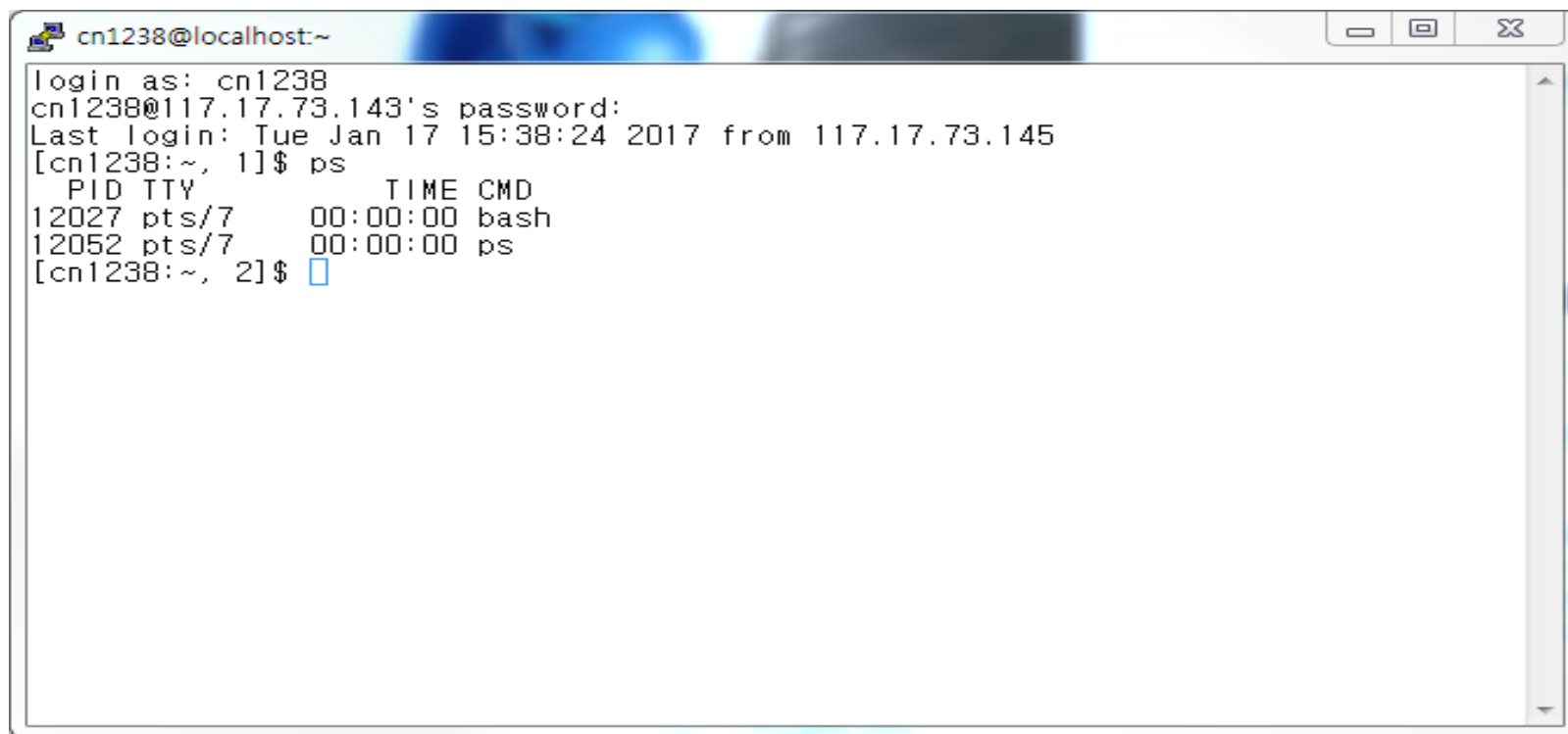
# 프로세스 관리 실무

---

# 프로세스 관리 실무

## 프로세스 확인

- Ps
  - 프로세스의 상태 확인



```
cn1238@localhost:~  
login as: cn1238  
cn1238@117.17.73.143's password:  
Last login: Tue Jan 17 15:38:24 2017 from 117.17.73.145  
[cn1238:~, 1]$ ps  
  PID TTY          TIME CMD  
 12027 pts/7        00:00:00 bash  
 12052 pts/7        00:00:00 ps  
[cn1238:~, 2]$
```

# 프로세스 관리 실무

## 프로세스 확인

- Ps -ef | grep 프로세스명

```
[cn1238:~, 8]$ ps -ef | grep bash
ROOT      5551    5549    0 Jan16 pts/2      00:00:00 bash
cn1238    6048    6046    0 Jan16 pts/5      00:00:00 bash
admin     6540    6538    0 Jan16 pts/9      00:00:00 bash
root      6557    6554    0 Jan16 pts/9      00:00:00 bash
cn1447    9704      1    0 2016 ?          00:00:00 -bash
cn1238    21634   21633    0 10:32 pts/1      00:00:00 -bash
cn1238    21816   21634    0 10:47 pts/1      00:00:00 grep bash
```

# 프로세스 관리 실무

## 프로세스 트리 확인

- Pstree

```
[cn1238:~, 9]$ pstree
init--NetworkManager
    |--abrt-d
    |--acpid
    |--atd
    |--auditd--{auditd}
    |--automount--4*[{automount}]
    |--avahi-daemon--avahi-daemon
    |--bash
    |--4*[bonobo-activat]--{bonobo-activat}
    |--3*[clock-applet]
    |--console-kit-dae--63*[{console-kit-da}]
    |--crond
    |--cupsd
    |--5*[dbus-daemon]
    |--7*[dbus-launch]
    |--devkit-power-da
    |--3*[gconf-im-settin]
    |--4*[gconfd-2]
    |--gdm-binary--gdm-simple-slav--Xorg
        |--gdm-session-wor--gnome-session--abrt-ap+
        |                                       |--bluetoo+
        |                                       |--gdu-not+
        |                                       |--gnome-p+
        |                                       |--gnome-p+
        |                                       |--gnome-v+
        |                                       |--gpk-upd+
        |                                       |--metacity
        |                                       |--nautilus
        |                                       |--nm-appl+
        |                                       |--polkit-+
        |                                       |--python
        |                                       |--{gnome-+
        |--2*[gdm-simple-slav--Xorg]
            |--gdm-session-wor--gnome-session--abrt+
            |                                       |--blue+
            |                                       |--gdu-+
            |                                       |--gnom+
            |                                       |--gnom+
            |                                       |--gnom+
            |                                       |--gpk-+
            |                                       |--meta+
            |                                       |--naut+
            |                                       |--polk+
            |                                       |--pyth+
            |                                       |--{gno+
```

# 프로세스 관리 실무

## 프로세스 트리 확인

### • Pstree -up 사용자이름

```
[cn1238:~, 10]$ pstree -up cn1238
bonobo-activati(5812)---{bonobo-activat}(5813)
clock-applet(5898)
dbus-daemon(5696)
dbus-launch(5695)
gconf-im-settin(6023)
gconfd-2(5769)
gnome-session(5686)---abrt-applet(5818)
                        ---bluetooth-apple(5824)
                        ---gdu-notificatio(5833)
                        ---gnome-panel(5808)
                        ---gnome-power-man(5830)
                        ---gnome-volume-co(5822)
                        ---gpk-update-icon(5819)---{gpk-update-ico}(8380)
                        ---metacity(5803)
                        ---nautilus(5810)
                        ---polkit-gnome-au(5820)
                        ---python(5832)
                        ---{gnome-session}(5771)
```

# 프로세스 관리 실무

## 진행 프로세스

- jobs
  - 현재 진행중이거나 멈춘 프로세스 표시

```
[root@host backup]# jobs
[1]-  Stopped
[2]+  Stopped
```

```
tar -cvf boot.tar.gz boot
tar -cvf boot.tar.gz boot
```



# 프로세스 관리 실무

## 백그라운드

- 명령어 &
  - 명령어 행 마지막에 &를 붙이면 백그라운드 실행

```
[root@host backup]# tar -cf boot.tar.gz boot &  
[1] 7907  
[root@host backup]# jobs  
[1]+  Done                  tar -cf boot.tar.gz boot
```

# 프로세스 관리 실무

## 백그라운드

- Ctrl + Z
  - 현재 진행중인 프로세스 일시중지
- bg %번호
  - 일시정지된 프로세스를 백그라운드에서 실행

```
^Z
[2]+  Stopped                  tar -cvf boot.tar.gz boot
[root@host backup]# jobs
[1]-  Stopped                  tar -cvf boot.tar.gz boot
[2]+  Stopped                  tar -cvf boot.tar.gz boot
[root@host backup]# bg%1
bash: bg%1: command not found
[root@host backup]# bg %1
[1]-  tar -cvf boot.tar.gz boot &
[root@host backup]# jobs
[1]-  Done                     tar -cvf boot.tar.gz boot
[2]+  Stopped                  tar -cvf boot.tar.gz boot
```

# 프로세스 관리 실무

## 포그라운드

- fg %번호

- 일시정지된 프로세스를 포그라운드에서 실행

```
[root@host backup]# jobs
[1]-  Done
[2]+  Stopped
[root@host backup]# fg %2
tar -cvf boot.tar.gz boot
[root@host backup]# ls
boot  boot.tar.gz  z
[root@host backup]# jobs
[root@host backup]#
```

```
tar -cvf boot.tar.gz boot
tar -cvf boot.tar.gz boot
```

# 프로세스 관리 실무

## 프로세스 제거

- Kill -9 PID
  - 해당 프로세스를 제거
  - -9는 프로세스를 제거하라는 시그널

```
[root@host backup]# kill 8135
```

```
[root@host backup]# ps
```

PID	TTY	TIME	CMD
6984	pts/4	00:00:00	su
6991	pts/4	00:00:00	bash
8135	pts/4	00:00:00	tar
8139	pts/4	00:00:00	tar
8266	pts/4	00:00:00	ps

```
[root@host backup]# kill -9 8135
```

```
[1]- Killed tar -cvf boot.tar.gz boot
```

# 프로세스 관리 실무

## 프로세스 제거

- Killall 프로세스명
  - 해당 프로세스들을 모두 제거
  - -i 실행할지 확인 옵션

```
[root@host backup]# jobs
[1]+  Running                  shutdown -h +100 &
[root@host backup]# killall shutdown
shutdown: Shutdown cancelled
[1]+  Done                      shutdown -h +100
```

# 퍼미션과 SetUID, 속성관리 실무

---

# 17장. 퍼미션과 SetUID,속성관리 실무

## 1. 파일에 대한 허가권의 종류와 기본 개념

**파일의 허가권** : 파일이나 디렉토리가 읽혀지거나 쓰여지거나 실행되려고 할 때, 그리고 옮겨지거나 삭제되는 등의 action 이 발생하려고 할 때 그 action에 대한 허가를 설정 하고 있는 것

→action 에 대한 허가:

- 1) 퍼미션
- 2) SetUID,SetGID,StickyBit
- 3) 속성

# 17장. 퍼미션과 SetUID ,속성관리실무

## 2. 파일과 디렉토리의 퍼미션

chmod : 특정파일 또는 디렉토리의 퍼미션을 변경하는 명령어

### <파일과 디렉토리의 퍼미션(권한)의 의미>

퍼미션	일반적인 의미	파일	디렉토리
r	읽기권한	파일을 읽을 수 있음	디렉토리의 내용을 볼 수 있음(ls)
w	쓰기권한	파일에 저장,삭제 할 수 있음	디렉토리에 파일저장, 디렉토리 이름변경,삭제
x	실행권한	파일 실행 가능	디렉토리에 접근할 수 있음(cd)
s	SetUID, SetGID	파일소유자(SetUID) , 그룹소유자(SetGID)의 권한으로 실행함	
t	Sticky Bit	공유디렉토리로 사용됨	



# 17장. 퍼미션과 SetUID , 속성관리실무

## 2-1. 파일의 퍼미션의 이해와 설정법

```
[root@localhost ~]# ls -l
```

```
total 60
```

```
-rw----- 1 root root 2671 Jan 18 05:54 anaconda-ks.cfg
-rw-r--r-- 1 root root 38587 Jan 18 05:54 install.log
-rw-r--r-- 1 root root 10067 Jan 18 05:53 install.log.syslog
```

-                      rw-                      ---                      ---  
 (파일 유형)    (소유자권한)    (그룹소속자권한)    (일반다른사용자권한)

\*파일유형: b: 블럭구조의 특수파일  
               c: 입출력에 사용되는 특수파일  
               p: 파이프파일임  
               s: 소켓파일임  
               -: 일반파일  
               l: 심벌릭링크파일  
               d: 디렉토리

# 17장. 퍼미션과 SetUID, 속성관리실무

## 3. SetUID 에 대한 이해와 설정법

### -SetUID가 설정된 파일을 실행하였을 경우:

실행되는 동안에는 실행시킨 사용자의 권한이 아닌 파일의 소유자 권한으로 실행된다 (=실행시작부터 실행 종료 할 때 까지 실행시킨 사용자의 권한이 아닌 그 파일의 소유자 권한으로 실행이 된다)

SetUID 퍼미션 설정 방법: 일반 퍼미션 앞에 4를 붙인다

```
[root@localhost ~]# ls -l install.log
-rw-r--r--. 1 root root 38587 Jan 18 05:54 install.log
[root@localhost ~]# chmod 4755 install.log
[root@localhost ~]# ls -l install.log
-rwsr-xr-x. 1 root root 38587 Jan 18 05:54 install.log
```

→ 소유자 실행퍼미션 자리에 s라는 문자가 설정됨  
다른 사용자들이 이 파일을 실행하더라도 파일의 소유자 권한으로 실행됨

# 17장. 퍼미션과 SetUID , 속성관리 실무


## 3. SetGID 에 대한 이해와 설정법

-SetGID 설정 파일 실행:

어떤 누가 실행하더라도 그 파일의 그룹권한으로 실행된다

SetGID 설정 방법: 일반 퍼미션 앞에 2를 붙인다

```
[root@localhost ~]# ls -l install.log
-rwsr-xr-x. 1 root root 38587 Jan 18 05:54 install.log
[root@localhost ~]# chmod 2755 install.log
[root@localhost ~]# ls -l install.log
-rwxr-sr-x. 1 root root 38587 Jan 18 05:54 install.log
```



그룹 소유자의 실행퍼미션 자리에 s로 설정되면 SetGID 로 설정된 것

# 17장. 퍼미션과 SetUID, 속성관리실무

## 3. Sticky Bit에 대한 이해와 설정법

```
[root@localhost ~]# ls -l / | grep tmp
- drwxrwxrwt. 19 root root 4096 Jan 22 21:54 tmp
```

일반 퍼미션: 777, 맨 끝에 t라는 소문자 마킹  
공유디렉토리: 파일생성을 아무나 할 수 있으며 생성된 파일은 각  
각 생성한 소유자의 소유로 되며 수정, 삭제 또한 생성한 사용자만  
이 그 파일을 수정하고 지울 수 있도록 되어있다  
Sticky Bit 설정 방법: 일반 퍼미션 앞에 1을 붙임

```
[root@localhost ~]# chmod 1777 test
[root@localhost ~]# ls -l
total 60
-rw----- 1 root root 2671 Jan 18 05:54 anaconda-ks.cfg
-rwxr-sr-x 1 root root 38587 Jan 18 05:54 install.log
-rw-r--r-- 1 root root 10067 Jan 18 05:53 install.log.syslog
-rwxrwxrwt 1 root root 0 Jan 22 22:50 test
```

실행 퍼미션자리에 t라는 문자로 셋팅  
즉, test파일이 공유 목적으로 사용 가능

★ Sticky Bit가 설정된 디렉토리내의 파일들에 대하여 root는 언제든지  
마음대로 변경, 삭제등이 가능하다

# 17장. 퍼미션과 SetUID , 속성관리실무

## 4. 파일과 디렉토리의 속성에 대한 이해와 활용

-chattr: 특정 속성을 부여하거나 제거할 수 있는 명령어  
-lsattr: 파일에 부여되어있는 특정 속성을 확인하는 명령어  
<속성>

- 1) a속성: 해당 파일을 추가만 할 수 있다
- 2) c속성: 커널에 의해 디스크상에 자동적으로 압축된 상태
- 3) d 속성: dump 프로그램(명령어)으로 백업되지 않는다
- 4) i 속성: 해당파일의 변경,삭제,파일추가,링크파일도 만들 수 없다
- 5) s 속성: 파일삭제시 해당블럭이 모두 0, 디스크에 다시 쓰기가 발생함
- 6) S 속성: 파일 변경시 디스크동기화 발생
- 7) u 속성: 삭제시 그 내용이 저장되며 삭제 전 데이터로 복구 가능
- 8) j 속성: 저널링 파일시스템의 파일을 대상으로 설정
- 9) A 속성:access 가 발생하더라도 저장되는 atime값은 갱신되지 않음
- 10)D 속성: 변경사항을 동기화시킴
- 11)T 속성: 최상위 디렉토리로 인식되어 간주됨

✓ Chattr로 파일가 디렉토리의 속성을 지정하는 주된 이유는 허가되지 않은 사용자가 파일의 변경을 못하도록 설정하여 파일보안을 하기 위해서이다

# 17장. 퍼미션과 SetUID , 속성관리실무

## 4. 속성 부여하기와 제거하기

- 속성 부여하기: `chattr +속성 파일이름`
- 속성 제거하기: `lsattr -속성 파일이름`
- 여러 개의 속성 동시에 부여하기: 각각의 속성은 독립적이기 때문에 여러 개의 속성을 동시에 설정 가능 +원하는 속성 이어서 사용

```
[root@localhost ~]# ls -l
total 60
-rw-----. 1 root root 2671 Jan 23 2017 anaconda-ks.cfg
-rw-r--r--. 1 root root 38587 Jan 23 2017 install.log
-rw-r--r--. 1 root root 10179 Jan 23 2017 install.log.syslog
[root@localhost ~]# ls -l install.log
-rw-r--r--. 1 root root 38587 Jan 23 2017 install.log
[root@localhost ~]# lsattr install.log
-----e- install.log
[root@localhost ~]# chattr +aisSu messages
chattr: No such file or directory while trying to stat messages
[root@localhost ~]# chattr +aisSu install.log
[root@localhost ~]# lsattr install.log
suS-ia-----e- install.log
```

Install.log 파일에 a i s S u 속성 동시 설정

# 17장. 퍼미션과 SetUID , 속성관리실무

## 4. 파일(디렉토리)의 소유자 및 소유그룹 변경하기 실무

chown(Change OWNeR) : 파일이나 디렉토리의 소유자와 소유그룹을 변경할 때에 사용하는 명령어

<소유자를 변경하고 싶을 때> chown 변경할 소유자 이름 파일이름

예) chown bible test : test 파일의 소유자를 bible로 변경

<파일의 소유자와 소유그룹 동시 변경> chown 소유자:소유그룹 파일이름

예) chown bible1:bible2 test : test 파일의 소유자를 bible1으로, 소유그룹을 bible2로 변경함

# 로그인과 원격접속

---



# 18장. 로그인과 원격접속

## 가장 기본적인 로컬시스템

/bin/login

```
[root:~, 14]# strings /bin/login
/lib/ld-linux.so.2
Wy7!
W+F|
PTRh
d$$[]
[^_]
T$4u
|$,1
execf
d$, [^_]
d$, [^_]
[^_]
LOGIN FAILURE FROM %s, %s
LOGIN FAILURE ON %s, %s
```

모든 로그인은 /bin/login 이라는 로그인과정을 전담하고 있는  
프로세스에 의해서 진행

# 18장. 로그인과 원격접속

가장 기본적인 로컬시스템

`/etc/securetty`

```
[root:~, 16]# cat /etc/securetty  
console  
vc/1  
vc/2  
vc/3  
vc/4  
vc/5  
vc/6  
vc/7  
vc/8  
vc/9
```

Root에 대한 로그인이 허용되는 터미널 장치명이 설정되어있는 파일

# 18장. 로그인과 원격접속

## 가장 기본적인 로컬시스템

### /var/log/secure, /var/log/messages

```
[root:~, 18]# tail -f /var/log/secure
Jan 23 14:08:07 localhost login: pam_unix(remote:auth): authentication failure; logname= uid=0 euid=0 tty=pts/6 ruser= rhost=171.249.29.65 user=root
Jan 23 14:08:09 localhost login: FAILED LOGIN 2 FROM 171.249.29.65 FOR root, Authentication failure
Jan 23 14:08:09 localhost login: pam_securetty(remote:auth): access denied: tty 'pts/6' is not secure !
Jan 23 14:08:11 localhost login: FAILED LOGIN 3 FROM 171.249.29.65 FOR root, Authentication failure
Jan 23 14:08:11 localhost login: pam_securetty(remote:auth): access denied: tty 'pts/6' is not secure !
Jan 23 14:08:11 localhost login: pam_unix(remote:auth): check pass; user unknown
Jan 23 14:08:11 localhost login: pam_unix(remote:auth): authentication failure; logname= uid=0 euid=0 tty=pts/6 ruser= rhost=171.249.29.65
Jan 23 14:08:11 localhost login: pam_succeed_if(remote:auth): error retrieving information about user guest
Jan 23 14:08:13 localhost login: FAILED LOGIN SESSION FROM 171.249.29.65 FOR guest, User not known to the underlying authentication module
Jan 23 14:08:13 localhost login: PAM 1 more authentication failure; logname= uid=0 euid=0 tty=pts/6 ruser= rhost=171.249.29.65 user=root
^C
[root:~, 19]# tail -10 /var/log/messages
Jan 23 14:08:02 localhost xinetd[1667]: START: telnet pid=24366 from::ffff:171.249.29.65
Jan 23 14:08:13 localhost xinetd[1667]: EXIT: telnet status=0 pid=24366 duration=11(sec)
Jan 23 14:08:16 localhost telnetd[24361]: ttloop: peer died: EOF
Jan 23 14:08:16 localhost xinetd[1667]: EXIT: telnet status=1 pid=24361 duration=32(sec)
Jan 23 14:08:19 localhost xinetd[1667]: START: telnet pid=24368 from::ffff:68.43.9.85
Jan 23 14:08:50 localhost xinetd[1667]: START: telnet pid=24370 from::ffff:68.43.9.85
Jan 23 14:08:50 localhost telnetd[24368]: ttloop: peer died: EOF
Jan 23 14:08:50 localhost xinetd[1667]: EXIT: telnet status=1 pid=24368 duration=31(sec)
Jan 23 14:09:21 localhost telnetd[24370]: ttloop: peer died: EOF
Jan 23 14:09:21 localhost xinetd[1667]: EXIT: telnet status=1 pid=24370 duration=31(sec)
```

위의 두 경로에서 로그인 관련 기록 확인 가능  
(단, messages경로는 접속기록뿐 아니라 시스템전체 로그가 기록됨)

# 18장. 로그인과 원격접속

## ssh를 이용한 원격접속

### ps -ef | grep ssh

```
[root:~, 24]# ps -ef | grep ssh
root      1659      1    0   2016 ?        00:00:00 /usr/sbin/sshd
root      21629    1659    0  10:32 ?        00:00:00 sshd: cn1238 [priv]
cn1238    21633    21629    0  10:32 ?        00:00:00 sshd: cn1238@pts/1
root      24079    1659    0  13:46 ?        00:00:00 sshd: root@pts/4
root      24420    24081    0  14:17 pts/4    00:00:00 grep ssh
[root:~, 26]# /etc/rc.d/init.d/sshd restart
Stopping sshd:                                     [ OK ]
Starting sshd:                                     [ OK ]
[root:~, 27]# ps -ef | grep ssh
root      21629      1    0  10:32 ?        00:00:00 sshd: cn1238 [priv]
cn1238    21633    21629    0  10:32 ?        00:00:00 sshd: cn1238@pts/1
root      24079      1    0  13:46 ?        00:00:00 sshd: root@pts/4
root      24435      1    0  14:18 ?        00:00:00 /usr/sbin/sshd
root      24438    24081    0  14:18 pts/4    00:00:00 grep ssh
```

ssh서버에는 sshd라는 데몬이 실행되어 있어야만  
다른 곳에서 이 서버로 ssh원격 접속이 가능하다

# 18장. 로그인과 원격접속

## ssh를 이용한 원격접속

### /etc/ssh/ssh\_config

```
[root:~, 28]# cat /etc/ssh/ssh_config
# $OpenBSD: ssh_config,v 1.25 2009/02/17 01:28:32 djm Exp $

# This is the ssh client system-wide configuration file. See
# ssh_config(5) for more information. This file provides defaults for
# users, and the values can be changed in per-user configuration files
# or on the command line.

# Configuration data is parsed as follows:
# 1. command line options
# 2. user-specific file
# 3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options. For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

# Host *
# ForwardAgent no
# ForwardX11 no
# RhostsRSAAuthentication no
# RSAAuthentication yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
```

### 클라이언트측 설정파일

# 18장. 로그인과 원격접속

## ssh를 이용한 원격접속

### /etc/ssh/sshd\_config

```
[root:~, 29]# cat /etc/ssh/sshd_config
# $OpenBSD: sshd_config,v 1.80 2008/07/02 02:24:18 djm Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/bin:/usr/bin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options change a
# default value.

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

# Disable legacy (protocol version 1) support in the server for new
# installations. In future the default will change to require explicit
# activation of protocol 1
Protocol 2

# HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
# HostKeys for protocol version 2
```

### 서버측 설정파일

# 18장. 로그인과 원격접속

ssh를 이용한 원격접속

ssh IP주소

```
[root:~, 30]# ssh 117.17.73.201  
root@117.17.73.201's password:  
Last login: Sun Jan 22 21:11:39 2017 from 117.17.73.205  
[root@localhost ~]#
```

ssh을 이용하여 원격접속  
(143에서 201로)

# 18장. 로그인과 원격접속

ssh를 이용한 원격접속

ssh -l IP주소

```
[root:~, 34]# ssh -l root 117.17.73.201  
root@117.17.73.201's password:  
Last login: Sun Jan 22 21:27:20 2017 from 117.17.73.143  
[root@localhost ~]#
```

지정한 사용자로 원격접속  
(143에서 201로)



# 18장. 로그인과 원격접속

## telnet을 이용한 원격접속

### telnet IP주소

```
[root:~, 20]# telnet 117.17.73.201
Trying 117.17.73.201...
Connected to 117.17.73.201.
Escape character is '^]'.
CentOS release 6.1 (Final)
Kernel 2.6.32-131.0.15.el6.i686 on an i686
login: dasol
Password:
Last login: Sun Jan 22 22:19:17 from 117.17.73.143
[dasol@localhost ~]$
```

telnet을 이용하여 원격접속  
(143에서 201로)

# 18장. 로그인과 원격접속

telnet을 이용한 원격접속

telnet -l

```
[root:~, 48]# telnet -l dasol 117.17.73.201  
Trying 117.17.73.201...  
Connected to 117.17.73.201.  
Escape character is '^]'.  
Password:  
Last login: Sun Jan 22 22:19:34 from 117.17.73.143  
[dasol@host ~]$
```

지정한 계정명으로 원격접속

# 18장. 로그인과 원격접속

## 원격파일복사

scp

```
[ds1556:~, 5]$ ls
1      ass_t22.c  book.c
2      ass_t23.c  dada
```



```
[root@localhost ~]# ls
anaconda-ks.cfg  install.log  install.log.syslog  test  test3
[root@localhost ~]# scp 117.17.73.143:/home/ds-51/ds1556/dada .
root@117.17.73.143's password:
dada                                     100%   0   0.0KB/s   00:00
[root@localhost ~]# ls
anaconda-ks.cfg  dada  install.log  install.log.syslog  test  test3
```

## 원격파일복사

# 18장. 로그인과 원격접속

## 원격파일복사

scp -r

```
[ds1556:~/winter, 13]$ ls
a.out prac1.c prac2.c prac3.c prac4.c prac5.c prac6.c tree1.c
[ds1556:~/winter, 14]$ pwd
/home/ds-51/ds1556/winter
```



```
[root@localhost ~]# ls
anaconda-ks.cfg dada install.log install.log.syslog test test3
[root@localhost ~]# scp -r 117.17.73.143:/home/ds-51/ds1556/winter .
root@117.17.73.143's password:
prac3.c                100% 304      0.3KB/s   00:00
prac1.c                100% 143      0.1KB/s   00:00
prac5.c                100% 410      0.4KB/s   00:00
a.out                  100% 5160     5.0KB/s   00:00
tree1.c                100% 2447     2.4KB/s   00:00
prac4.c                100% 168      0.2KB/s   00:00
prac6.c                100% 386      0.4KB/s   00:00
prac2.c                100% 132      0.1KB/s   00:00
[root@localhost ~]# ls
anaconda-ks.cfg dada install.log install.log.syslog test test3 winter
```

원격파일복사(디렉토리)

# 18장. 로그인과 원격접속

## 원격로그인

slogin

```
[root@localhost ~]# slogin -l ds1556 117.17.73.143  
ds1556@117.17.73.143's password:  
Last login: Mon Jan 23 16:10:35 2017 from 117.17.73.201  
[ds1556:~, 1]$
```

원격로그인