



# 리눅스 구조

2007. 10. 24

안효창

# Contents

- 1 파일시스템과 볼륨
- 2 마운트
- 3 RPM
- 4 부팅과정과 시스템 초기화
- 5 시스템 설정 정보

# 파일시스템과 볼륨

- 파일시스템
  - 파일의 실제 데이터와 메타데이터를 유지/저장하는 체계
  - 각 운영체제별 고유의 파일시스템이 존재
    - Dos → FAT16
    - Windows 9x → FAT32
    - Windows NT 계열 → NTFS
    - 리눅스 → ext2, ext3
  - ext2 단점
    - 무결성 검사(fsck)에 시간이 오래걸림
    - 비정상 종료시 데이터 손실의 가능성이 다분함

# 파일시스템과 볼륨

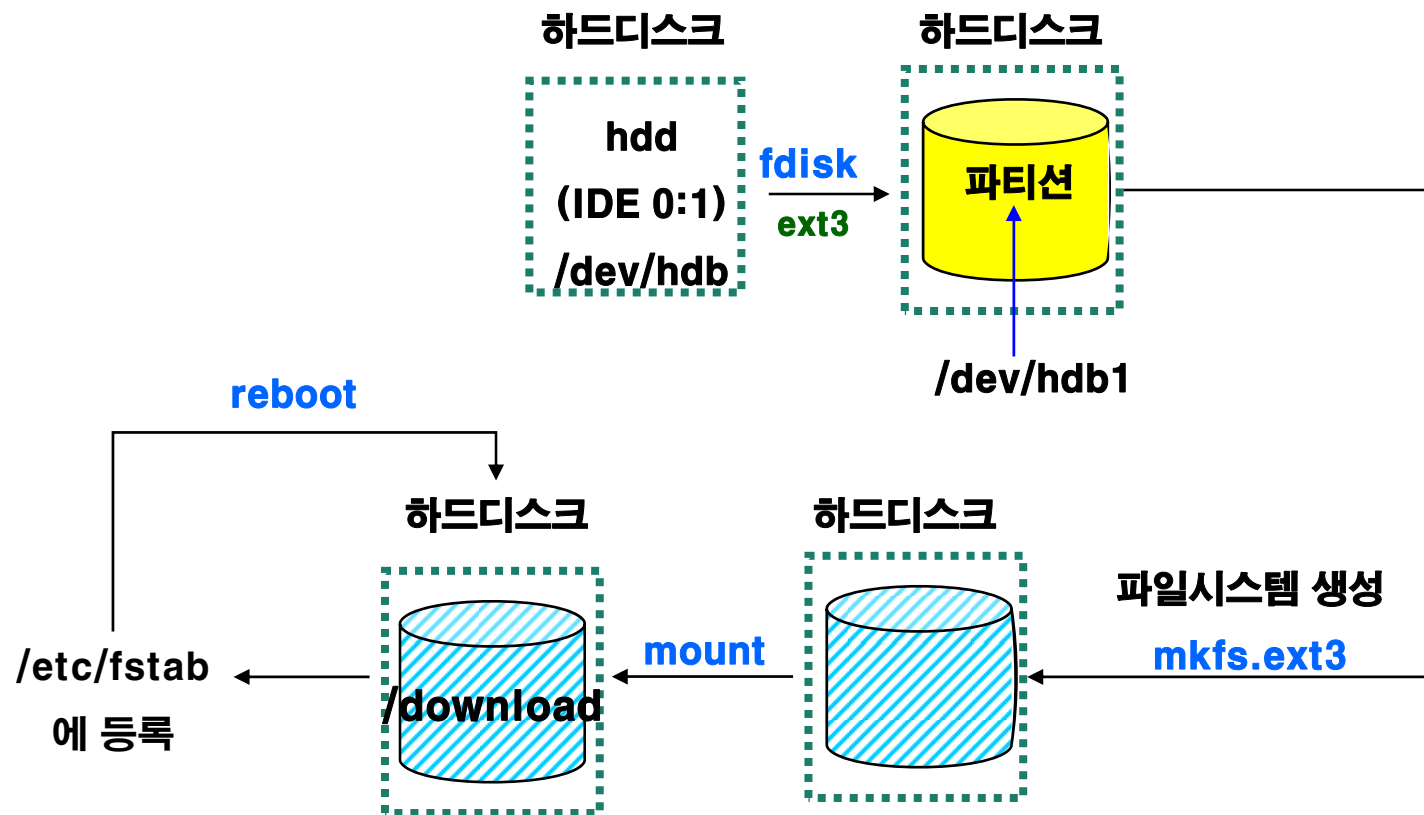
- 저널링 파일시스템
  - 백업 및 복구 능력이 뛰어남
  - 복구시간의 단축 및 데이터 무결성 보장
  - ext2에서 ext3로의 변환이 쉬움
  - $\text{ext3} = \text{ext2} + \text{저널링}$
  - Redhat9는 ext3를 기본 파일시스템으로 사용.

# 파일시스템과 볼륨

- 하드디스크 추가
  - fdisk /dev/hdb
    - Command : n
    - Command action : p
    - Patition number (1-4) : 1
    - First cylinder
    - Last cylinder
    - Command : p
    - Command : w
  - mkfs.ext3 /dev/hdb1
  - mount -t ext3 /dev/hdb1 /data
  - umount /data/
- /etc/fstab

# 파일시스템과 볼륨

- HDD 추가 흐름도



# 마운트

- 마운트
  - CD 장치나 하드디스크 등의 물리적 장치를 특정 디렉토리에 연결하는 명령어
  - Windows 계열
    - vfat, ntfs, iso9660 파일시스템을 설치단계에서 결정 후, 계속 사용
    - 제한된 사용이지만, 사용자가 별도 신경쓸게 없음
  - 리눅스
    - 다양한 파일시스템을 수시로 연결하고 해제
    - 여러 파일시스템에 다양하게 접근 가능하지만, 불편
    - /etc/fstab 에 등록할 경우, 부팅시 마다 자동으로 연결

# 마운트

## ■ 사용법

- `mount -t <파일시스템 타입> <장치명> <마운트 포인트>`
  - Cd-rom 의 경우
  - `# mount -t iso9660 /dev/cdrom /mnt/cdrom`
  - `# mount -t vfat /dev/fd0 /mnt/floppy`
  - mkisofs
  - Cdrecord
  - growisofs
- X윈도우에서는 자동으로 CD-Rom과 FDD를 자동으로 마운트 시켜줌
- 마운트 확인
  - `# mount` , `# df` , `# cat /etc/mstab`
- 마운트 해제
  - `# umount <장치명> 또는 <마운트 포인트>`



# RPM

- RPM 패키지의 탄생배경
  - Redhat Package Managemer
  - 리눅스의 설치가 어려운 점을 개선하여, 레드햇사에서 프로그램 설치 후에 바로 실행할 수 있는 패키지를 제작
  - 확장명: rpm
  - rpm 명령어를 통해 설치, 업그레이드, 질의, 검증, 삭제 등이 가능

# RPM

- RPM패키지의 영역별 의미
  - 패키지이름-버전-릴리즈 번호.아키텍처.rpm
    - 예 : mozilla-1.2.1-26.i386.rpm
  - 패키지이름 : 프로그램의 명칭
  - 버전 : 대개 3자리수로 구성되며, 높을 수록 상위 버전
  - 릴리즈 번호 : 패키지의 문제점 개선번호
  - 아키텍처
    - i386,i486,i586,i686 : 인텔계열 CPU
    - alpha/sparc/ia64... : 해당 CPU
    - src : 소스패키지, 설치 후에 별도 컴파일 및 환경설정을 해줘야 함
    - noarch : 아키텍처의 영향을 안받음, 모든 CPU에서 작동

# RPM

- rpm 명령어
  - 설치
    - # rpm -ivh <패키지 파일명>
      - i : 설치(install)
      - v : 설치과정의 확인(verbose)
      - h : 설치진행의 “#”마크가 화면에 출력됨(hash)
  - 삭제
    - # rpm -e <패키지명>
  - 업데이트
    - # rpm -Uvh <패키지 파일명>
      - U : 패키지가 없다면 “i”와 같이 일반 설치, 패키지가 있다면 업  
그레이드

# RPM

- 패키지 질의
- # rpm -qa <패키지명>
  - 패키지가 설치되었는지 확인
  - 예) # rpm -qa mozilla 또는 # rpm -qa | grep "moz\*"
- # rpm -qf <파일의 절대 경로>
  - 설치된 파일이 어느 패키지에 포함된 것인지 확인
  - 예) # rpm -qf /sbin/ifconfig
- # rpm -ql <패키지명>
  - 특정 패키지에 어떤 파일들이 포함되었는지 확인
  - 예) # rpm -ql mozilla

# RPM

- # rpm -qi <패키지명>
  - 설치된 패키지의 상세 정보
  - 예) # rpm -qi mozilla
- # rpm -qip <파일의 절대 경로>
  - 설치할 패키지의 정보
  - 예) #rpm -qip /mnt/cdrom/mozilla-1.2.1-26.i386.rpm
- /var/lib/rpm
  - rpm 패키지 데이터베이스

# RPM

- 패키지 의존성
  - 프로그램을 분할 제작하기 때문에 발생
  - A라는 프로그램을 설치하기 위해서는 B라는 프로그램이 먼저 설치되어야 하는 현상 때문에 B가 설치되지 않았다면 A도 설치할 수 없음
  - A가 설치되지 않을 때, 기본적으로 B라는 패키지를 찾기가 어려움
  - `rpmdb-redhat-9-0.20030313.i386.rpm` 을 설치하면 해결

# RPM

- 패키지 강제설치
  - A라는 프로그램과 B라는 프로그램이 상호의존적
    - 의존성 있는 프로그램을 소스 컴파일로 설치한 경우
    - 특정 파일을 삭제하였을 경우
  - # rpm -ivh --force <패키지 파일명>
  - # rpm -ivh --nodeps <패키지 파일명>
  - # rpm -ivh --force --nodeps <패키지 파일명>

# RPM

- YUM (Yellowdog Update Manager)
  - “rpm” 명령의 패키지 의존성문제를 해결
  - 인터넷을 통하여 Fedora의 rpm 파일이 저장된 저장소 (Repository)에서, 자동 다운로드 하여 설치
  - 사용법
    - 설치 : yum install <패키지명>
    - 삭제 : yum remove <패키지명>
    - 정보확인 : yum info <패키지명>



# RPM

- 패키지 질의
  - 이미 설치된 파일이 어느 패키지에 포함되어 있는가?
    - # rpm -qf <파일 절대경로>
    - 예) # rpm -qf /sbin/ifconfig
  - 앞으로 설치하고자 하는 패키지의 정보는 ?
    - # rpm -qip <rpm 파일 절대경로>
    - 예) #rpm -qip /mnt/cdrom/mozilla-1.2.1-26.i386.rpm
  - 이미 설치된 패키지의 정보는 ?
    - # rpm -qi <패키지명>
    - 예) # rpm -qi mozilla
  - 내 시스템에 설치된 패키지의 이름을 조회하려면 ?
    - # rpm -qa <패키지명>
    - 예) # rpm -qa mozilla 또는 # rpm -qa | grep "moz\*"

# 부팅과정과 시스템 초기화

- 부팅과정 요약
  - 전원공급 → Power On Self Test → BIOS에 설정된 부팅장치 → MBR (Master Boot Record) → 부트로더(LILO,GRUB)
- LILO(리로) : Linux LOader
  - 전통적인 부트로더
  - /etc/lilo.conf 에 설정파일 저장
- GRUB(그룹) : GRand Unified Bootloader
  - 레드햇 7.x부터 기본 부트로더로 제공
  - LILO 의 단점을 보완
  - 사용이 편리
  - 부트정보가 올바르지 않아도 부팅시에 바로 수정가능
  - 윈도우 계열 외에 다른 OS와 멀티부팅 가능
  - 대화형 모드제공
  - /boot/grub/grub.conf 에 설정파일 저장

## 부팅과정과 시스템 초기화

- `/etc/grub.conf` 또는 `/boot/grub/grub.conf` 설정파일
  - `timeout` = 부팅시간(초)
    - 예) `timeout=10`
  - `default` 엔트리번호 : 기본값으로 부팅되도록 설정되어진 순서
    - 예) `default 0`
  - `fallback` 엔트리번호 : 부팅실패시 차순위 부팅순서
    - 예) `fallback 1`
  - `splashimg` (장치명, 파티션) 이미지파일
    - 예) `splashimg (hd0,0) /boot/grub/splash.xpm.gz`

# 부팅과정과 시스템 초기화

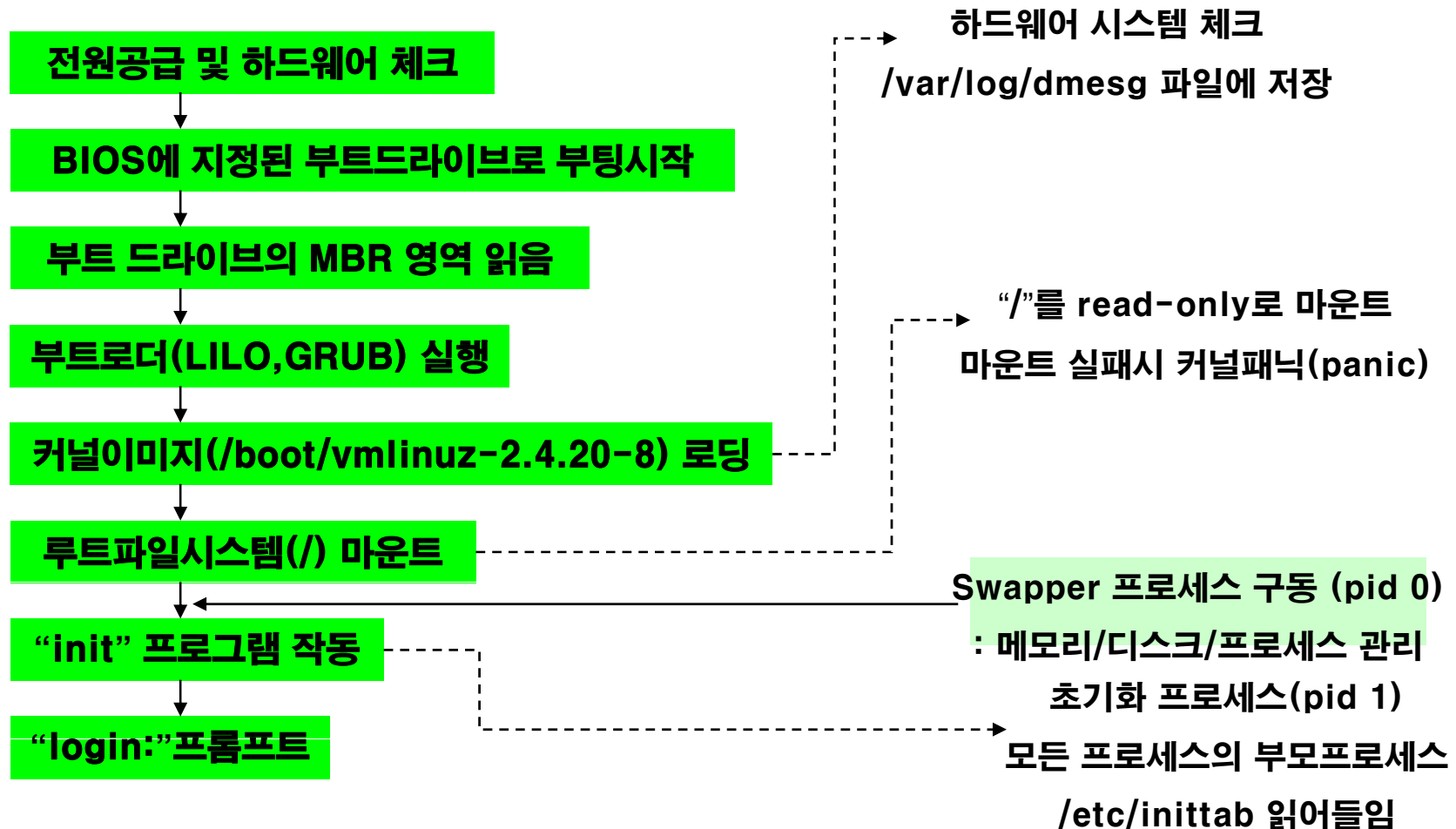
- title 부팅제목
  - 예) title Redhat 9 (2.4.20)
- root (하드디스크장치, 파티션)
  - 예) root (hd0,0)
- kernel /vmlinuz-버전 ro root=/dev/장치명
  - 예) kernel /vmlinux-2.4.20 ro root=/dev/hda2
- initrd 이미지파일명 : 시스템 초기화용 램디스크 이미지 파일

# 부팅과정과 시스템 초기화

- Windows와 멀티부팅
  - /etc/grub.conf 에 아래형식 추가
    - title 타이틀명
      - 예) title Windows XP Pro
    - rootnoverify (장치명, 파티션)
      - 예) rootnoverify (hd0,0)
    - makeactive
    - chainloader +1
  - /etc/grub.conf 설정이 잘못되어 부팅이 안될 때
    - GRUB부팅화면에서 “e”키로 해당 메뉴 편집

# 부팅과정과 시스템 초기화

## ■ 시스템 부팅 과정



# 부팅과정과 시스템 초기화

- 런레벨(runlevel)
  - 시스템을 운영하는 7개의 방법
  - Windows계열의 표준모드, 안전모드, 도스프롬프트모드 등과 비슷
  - 리눅스 부팅시 결정
  - /etc/rc.d/rc 파일 → /etc/rc?.d/ 디렉토리 파일 실행
  - 종류
    - 0 : 시스템 종료 (# halt == # init 0)
    - 1 : 싱글모드, root사용자의 패스워드 분실시
    - 2 : 네트워크기능 없는 다중사용자 모드
    - 3 : 일반적 형태(텍스트). 네트워크기능이 포함된 다중사용자 모드
    - 4 : 예비 레벨
    - 5 : 일반적 형태 (그래픽). runlevel 3과 거의 유사.
    - 6 : 재부팅 모드 (# reboot == # init 6)

# 부팅과정과 시스템 초기화

- /etc/inittab 파일
  - “코드:런레벨:행동:명령어” 형식
  - 주요 코드
    - id:3:initdefault:
      - 기본으로 runlevel 3번으로 부팅
    - si::sysinit:/etc/rc.d/rc.sysinit
      - 시스템 초기화를 위한 설정
    - l0:0:wait:/etc/rc.d/rc 0
    - l1:1:wait:/etc/rc.d/rc 1
      - 기본으로 세팅된 런레벨에 의해서 실행
    - ca::ctrlaltdel:/sbin/shutdown -t3 -r now
      - ctrl-alt-del 키보드 사용
      - “#” 주석처리 권장



# 부팅과정과 시스템 초기화

항목	의미	
코드	구별을 위한 항목	
런레벨	어느 런레벨에서 동작할 것인가를 결정	
행동	initdefault	시스템 구동 시 사용할 기본 런레벨
	boot	부팅 시 실행
	sysinit	부팅시 한번만 실행(boot 보다 선행됨)
	wait	프로세스 종료시까지 기다린 후 다음 작업 실행
	respawn	프로세스를 종료할 때마다 해당 작업 재실행
명령어	실행할 명령어	

## 부팅과정과 시스템 초기화

- 현재 런레벨 확인
  - 명령어 : # runlevel
- 시스템 부팅시 특정 서비스 실행
  - 명령어 : # ntsysv
  - /etc/init.d/ 디렉토리에 존재하는 파일 → /etc/sysconfig/  
참조

## 시스템 설정 정보

- /etc/sysconfig 디렉토리의 파일
  - /etc/init.d/ 의 초기화 스크립트가 필요시 참조하는 파일
  - 서비스 구현시 각종 옵션을 변수값으로 기록한 파일
  - /etc/init.d/ 파일을 수정하는 것보다 /etc/sysconfig/ 설정값을 변경하는 것이 시스템 유지보수에 편리
  - rpm 패키지로 설치하지 않았을 때, 파일이 존재하지 않을 수도 있음
  - 필요에 따라 변경

## 시스템 설정 정보

- /etc/sysconfig/amd
  - amd 데몬 실행시 옵션과 디렉토리 설정
- /etc/sysconfig/apmd
  - 시작/종료/변경 및 보류 등에대한 설정
- /etc/sysconfig/authconfig
  - 사용자 정보 및 인증 방법 설정
  - # authconfig 명령을 사용하거나, 직접 변경가능
- /etc/sysconfig/clock
  - 시간에 관련된 내용을 설정하는 영역
- /etc/sysconfig/desktop
  - X윈도우 가동시 사용할 데스크탑메니저 결정
  - “GNOME”이나 “KDE” 선택

## 시스템 설정 정보

- /etc/sysconfig/harddisk
  - 하드디스크 설정 (주의!)
- /etc/sysconfig/hwconf
  - 현 리눅스에서 어떠한 하드웨어를 인식하는지 참고용
- /etc/sysconfig/i18n
  - 시스템에게 현재 사용되는 언어를 알려줌
- /etc/sysconfig/init
  - 부팅과정에서 시스템 초기화시 출력될 화면 설정
- /etc/sysconfig/iptables
  - 방화벽 및 NAT 설정시 유용한 설정영역
- /etc/sysconfig/keyboard
  - 키보드 제어방법을 설정

## 시스템 설정 정보

- /etc/sysconfig/kudzu
  - 부팅시 검색할 하드웨어 영역 설정
- /etc/sysconfig/mouse
  - 마우스 정보 설정
- /etc/sysconfig/named
  - 네임 서버 구동시 사용할 옵션 지정
- /etc/sysconfig/network
  - 네트워크에 관련된 기본 값 설정
- /etc/sysconfig/network-scripts
  - 네트워크 장치에 대한 정보를 기록파일 및 명령어가 있는 디렉토리
- /etc/sysconfig/vncservers
  - Vnc서버를 구동하는 방법 설정



■ *Any Questions ?*

