



리눅스 파일관리

2007. 11. 07

안효창

Contents

- 1 사용자 관리와 비밀번호
- 2 파일 위치 검색
- 3 파일 압축과 관리
- 4 쿼터

사용자 관리와 비밀번호

- 사용자 및 그룹 생성 및 관리
 - useradd, groupadd 명령어 사용
- 리눅스 제공 표준 사용자 및 표준 그룹
 - /etc/passwd , /etc/group
 - UID : 사용자계정과 대응하는 시스템이 인식하는 고유한 숫자
 - GID : 그룹명과 대응하는 시스템이 인식하는 고유한 숫자
 - 사용자 : ID와 Password를 가지고 리눅스 사용권한을 가짐
 - 그룹 : 여러 사용자에게 같은 권한을 가질 수 있도록 묶는 역할

사용자 관리와 비밀번호

- 명령어를 이용한 사용자 관리
 - root관리자가 생성
 - 명령어 : `useradd new-id [옵션] [변경값]`
 - 예) 디렉토리 지정 `#useradd woo -d /home/woo`
 - 쉘 지정 `#useradd woo -s /bin/tcsh`
 - 그룹 지정 `#useradd woo -g mygroup`

옵션	의미
-G	추가할 사용자의 gid 지정(지정하지 않으면 uid와 동일)
-g	추가할 사용자를 특정 그룹에 포함시킴
-d	사용자의 홈 디렉토리 지정(지정하지 않을 경우 /home/계정이름)
-s	사용자 쉘 지정(지정하지 않을 경우 /bin/bash)
-u	추가할 사용자 uid 지정(지정하지 않을 경우 500번부터 순차적으로 할당)
-M	홈 디렉토리를 생성하지 않는 사용자 추가 시 사용

사용자 관리와 비밀번호

- 명령어 : `passwd user-id`
 - 예) `#passwd woo`
- 명령어 : `userdel [옵션] user-id`
 - 예) 디렉토리까지 삭제 `#userdel -r woo`
- 명령어 : `usermod [옵션] [변경값] user-id`
 - 예) 그룹변경 `# usermod -g users woo`
 - 디렉토리 변경 `# usermod -d /woo woo`
- 명령어 : `chage [옵션] [변경값] user-id`
 - 예) ID사용 만료일지정 `# chage -E 2004/02/01 woo`

사용자 관리와 비밀번호

- 사용자 추가 시 기본값 → /etc/login.defs
- skel 디렉토리 → /etc/skel/
- 명령어 실행 시 작동순서
 - 1. /etc/default/useradd 파일에 사용자 추가할 경우 적용할 사항을 점검
 - 2. /etc/passwd 에 사용자추가
 - 3. /etc/group 에 그룹이름추가
 - 4. /home 디렉토리 밑에 계정이름과 같은 이름의 디렉토리 생성(계정공간)
 - 5. /etc/skel 디렉토리의 파일내용을 계정공간으로 복사
 - 6. 패스워드는 나중에 passwd <계정이름> 으로 설정

사용자 관리와 비밀번호

- root사용자는 모든 패스워드 변경가능
- 일반사용자는 자기 패스워드만 변경
- /etc/shadow와 /etc/passwd 차이
 - : passwd는 모든사용자가 읽기 가능,
 - : shadow는 root 만 읽기 가능
- 사용자 계정 변경
 - # su → root 권한 획득
 - # su - → root 권한 획득 + root 환경설정
 - # su 사용자명 → 사용자 권한 획득

사용자 관리와 비밀번호

- /etc/passwd 파일의 구조
 - account:password:UID:GID:GECOS:directory:shell
 - account : 사용자이름
 - password : 암호(shadows파일로 대치될 경우 * 표시)
 - UID, GID : 사용자번호, 그룹번호
 - GECOS : 사용자정보(사용자 Full Name)
 - directory : 사용자의 홈디렉토리
 - shell : 사용자의 기본 shell
 - 계정앞에 “*” 표시하면 계정잠금
 - 첫 User나 Group의 UID,GID는 500번으로 시작

사용자 관리와 비밀번호

▪ /etc/shadow 파일구조

account:encryptedpassword:last:may:must:warn:expire:disable
:reserved

- account 계정 이름
- encrypted password 암호화된 패스워드
- last 최근 패스워드 변경일 (1970년 1월 1일 기준의 날짜 수)
- may 패스워드 변경 후, 재설정을 위한 대기일 수
- must 패스워드 유효기간
- warn 패스워드 변경 경고 시간
- expire 패스워드 만료시간 이후 계정을 사용할 수 없게 되는 기간
- disable 계정 사용 불가능 날짜 (1970년 1월 1일 기준의 날짜 수)
- reserved 예약된 영역

사용자 관리와 비밀번호

- 그룹생성
 - # groupadd -g gid <그룹이름>
- 그룹삭제
 - # groupdel <그룹이름>
- 그룹변경
 - # groupmod -n <new 그룹이름> <old 그룹이름>

사용자 관리와 비밀번호

- /etc/group 파일구조

group_name:password:GID:user-list

- group_name : 그룹명
- Password : 그룹 패스워드(필요 시 사용, 일반적으로 사용하지 않음)
- GID : 그룹 번호
- User-list : 그룹에 속한 User ID , 콤마(,)로 구분

- 사용자 확인 명령

- # whoami
- # who am I
- # id
- # groups

파일 위치 검색

- which <실행파일명>
 - PATH에 설정된 디렉토리만 검색
 - 절대경로를 포함한 위치 검색
- whereis <실행파일명>
 - 실행파일 및 소스,맨페이지 파일까지 검색
- locate <파일명>
 - 매우 빠르고 유용하지만 새로 설치된 파일들이 등록되어 있지 않을 때는 찾을 수 없음.
 - 파일정보에 대한 데이터베이스 갱신이 매우 중요
 - 수시로 updatedb 를 돌려 파일정보 갱신 - crontab 에 등록하여 자동으로 구동시킴

파일 위치 검색

- find [경로] [옵션] [조건] [action]
 - 옵션 : -name, -user(소유자), -newer(전,후), -perm(허가권)
 - Action : -print(GNU는 디폴트, Unix는 필수입력), -exec
 - 예) #find /etc -name “*conf*”
 - #find / -user woo
 - #find ~ -perm 644
 - #find . -name “*tar*” -exec rm -f {} \;
 - → find의 결과가 {}안으로 들어감, “\;”는 관례적 필수로 생각하면 됨.

파일 위치 검색

- `grep [옵션] <패턴> <filename> [files...]`
 - 파일 내에서 지정한 패턴과 일치하는 패턴이 있는 라인 출력
 - 옵션 : `-n`(행번호가 같이 출력) , `-i`(대소문자 구분 안함) , `-v`(패턴이 포함되지 않는 라인 출력)
 - 예) `grep linux /etc/*`
 - → `/etc/` 폴더의 `linux` 라는 단어가 포함된 파일과 라인 출력

파일 압축과 관리

- 파일 압축하기
 - # gzip 파일명 → 파일명.gz 이 생성
 - # bzip2 파일명 → 파일명.bz2 가 생성
 - # compress 파일명 → 파일명.Z 가 생성
- 파일 압축 풀기
 - # gzip -d 파일명.gz == # gunzip 파일명.gz
 - # bzip2 -d 파일명.bz2 == # bunzip2 파일명.bz2
 - # compress -d 파일명.Z == # uncompress 파일명.Z
- 기타 압축 명령
 - # zip 압축파일명 압축할 파일 → 파일명.zip 생성
 - # unzip 파일명.zip

파일 압축과 관리

- tar [동작] [옵션] [묶음파일명] [filelist or dir]
 - tar 는 압축명령이 아니며 여러 개의 파일을 한 개의 파일로 묶어주는 명령어
 - 동작
 - c (새로운 묶음)
 - x (묶인 파일을 풀어줌),
 - t (묶음을 풀기전에 묶인 경로를 보여줌)
 - 옵션 :
 - f (필수) - 묶음 파일명 지정
→ 원래 tar는 tape장치 백업이 기본(생략시 tape로)
 - v - visual
 - z - tar + gzip (GNU only)
 - j - tar + bzip2 (GNU only)

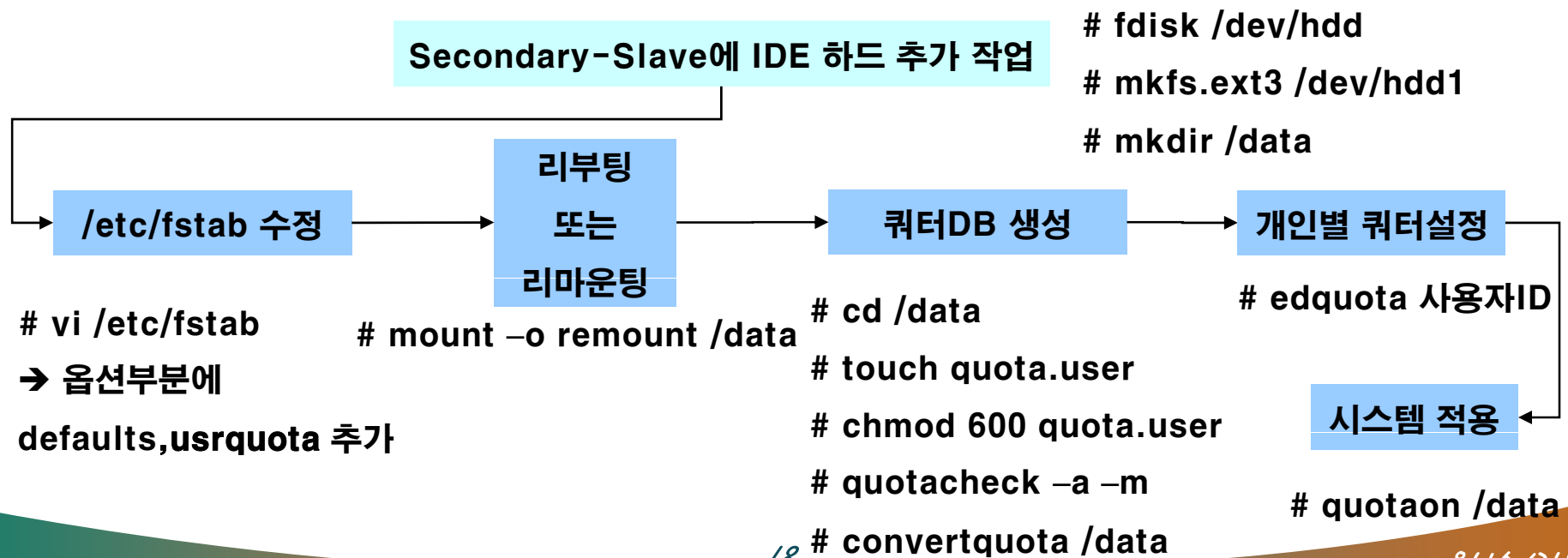
파일 압축과 관리

■ 예 :

- # tar cvf httpd.tar ./httpd → 생성
- # tar cvfz httpd.tar.gz ./httpd → 생성 + gzip압축
- # tar cvfj httpd.tar.bz2 ./httpd → 생성 + bzip2압축
- # tar tvf httpd.tar → 확인
- # tar xvf httpd.tar → 풀기
- # tar xvfz httpd.tar.gz → gzip압축해제 + tar풀기
- # tar xvfj httpd.tar.gz → bzip2압축해제 + tar풀기

쿼터

- 쿼터(quota)
 - 파일시스템마다 사용자나 그룹이 생성할 수 있는 파일의 용량 및 개수를 제한하는 것
- 적용 순서 요약
 - Data 백업을 위해 Secondary-Slave(/dev/hdd)에 HDD를 한 개 추가하고 , /data 디렉토리에 마운트 시킴.



쿼터

- 개인별 쿼터 설정
 - 명령어 : # edquota 사용자ID
 - vi와 사용법이 동일함

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/hdd1	0	0	0	0	0	0

현재 사용중인 블록(kb) 현재 사용중인 파일(개)

블록 사용 경고량(kb) 파일 사용 경고량(개)

블록 사용 제한량(kb) 파일 사용 제한량(개)

- 특정 사용자의 쿼터 설정을 다른 사용자에게 적용
 - # edquota -p <기존사용자> <새사용자>
- 기타
 - 쿼터 해제 : # quotaoff /data
 - 쿼터 설정 확인 : # repquota -a



■ *Any Questions ?*

