Report

서버관리실무

**제목 : 4주차 과제**

**학과 : 컴퓨터소프트웨어과  
학번 : 2017100301  
이름 : 이상남  
 담당교수 : 김형철 교수님  
제출일 : 2020.09.21**

1. **디렉터리 및 파일**
2. **cp(copy) : 파일이나 디렉터리를 복사하는 명령어이다.  
   [사용법] $ cp [options] source(s) destination  
   -r,-R: 복사하려는 대상이 디렉터리인 경우에 하위 디렉터리까지 같이 복사한다.  
   -I : 복사를 통해 생성할 파일이 존재할 경우에 덮어쓸 것인지를 물어본다  
   -f : 같은 파일의 이름을 존재하는 경우에 강제로 덮어쓴다.  
   -b : 덮어쓸 때 백업본의 파일을 만든다. 원래 파일명 뒤에 ~가 붙는 파일이 생성된다.  
   -s : 복사하는 대신에 심볼릭 링크를 만든다.  
   -p : 소유권, 그룹, 허가권등 모든 정보를 복사한다.  
   -v : 파일을 복사하기 전에 각각 파일명을 표시한다.  
   -d : 심볼릭파일인 경우 그 심볼릭 정보와 함께 링크파일을 그대로 복사.  
   -l : 복사대신에 하드 링크 파일을 만든다.  
   -u : 복사되는 대상이 최신의 파일이라면 복사되지 않는다.  
   [사용예]  
   $cp/etc/passwd . -> /etc/passwd라는 파일을 현재 디렉터리에 복사한다.  
   $cp /etc passwd passwd.txt ->/etc/passwd라는 파일을 현재 디렉터리에 passwd.txt라는 파일로 복사한다.  
   $cp -r /etc tmp -> /etc를 tmp 라는 디렉터리로 복사하는데, 하위에 있는 모든 파일과 디렉터리 까지 포함한다.  
   $cp a.txt b.txt /tep -> 현재 디렉터리의 a.txt b.txt를 /tmp 디렉터리로 복사한다.  
   $cp -r a.txt b.txt data/ data2/ ->현재디렉터리의 a.txt b.txt파일과 data 디렉터리를 data2디렉터리 안에 복사한다.**
3. **와일드카드  
   특정한 문자를 이용해서 임의의 문자열을 대체하도록 설정한 문자이다.  
   \* : all ? : 하나의 문자를 나타낸다**
4. **rm(remove) : 파일을 삭제하는 명령으로 옵션에 따라 디렉터리도 삭제한다.  
   [사용법] $rm [options] file(s) 또는 directory  
   -I : 파일을 삭제할 때 질의를 할 때 사용한다.  
   -f : 존재 유무나 어떠한 메시지 없이 강제로 지울 때 사용한다.  
   -r,-R : 하위디렉터리를 포함하여 디렉터리 안의 모든 내용을 지운다.  
   -- : 파일의 이름이 ‘-‘ 로 시작하는 파일을 지울 때 사용한다.  
   [사용예]  
   $ rm -I \* ->현재 디렉터리의 모든 파일을 하나씩 확이ㅣㄴ하면서 지운다.  
   $rm doc1.txt. doc2.txt do3.txt -> doc1.txt. doc2.txt do3.txt를 지운다.  
   $rm -rf data/ -> data라는 디렉터리와 그 안의 모든 파일과 디렉터리를 사용자 확인 메시지 없이 무조건 강재로 지운다.**
5. **mv(move) : 파일이나 디렉터리의 이름을 변경하거나 이동시키는 명령이다.  
   [사용법] : $mv [options] source(s) desination  
   -I : 이동할 디렉터리에 동일한 파일명이 존재할 경우에 덮어 쓸 것인지 물어본다.  
   -f : 같은 파일의 이름을 존재하는 경우에 강제로 덮어쓴다.  
   -b : 덮어쓸 때 백업본의 파일을 만든다.  
   -v : 파일을 이동하는 내용을 출력한다.  
   -u : 이동할 디렉터리에 동일한 파일명이 있을 경우에 검사하여 해당 파일이 최신이 파일이면 이동하지 않는다.  
   [사용 예]  
   $mv readme.txt /home/posein/data/ -> readme.txt 라는 파일을 /home/posein/data 디렉터리로 옮긴다.  
   $mv /home/posein/data/ /home/posein/backup/ -> /home/posein/data/라는 디렉터리를 /home/posein/backup/으로 이름을 변경한다.  
   $mv aaa.txt bbb.txt ->aaa.txt파일의 이름을 bbb.txt로 변경한다.  
   $mv a.txt b.txt data/ /tmp -> a.txt, b.txt라는 파일과 data라는 디렉터리를 /tmp디렉터리로 옮긴다.  
   $mv data/\* backup/ ->data디렉터리안의 파일 및 디렉터리만 backup디렉터리로 이동한다.**
6. **touch : 파일의 최종 접근 시간, 수정시간 등 타임 스탬프(Timestamp)를 변경하거나 파일의 크기가 0인 빈(empty) 파일을 생성하는 명령이다.  
   [사용법] : $touch [option] 파일명   
   -a : 접근시간(access time)을 바꾼다.  
   -m : 수정시간(modification time)을 바꾼다.  
   -t : 지정된 타임스탬프로 변경한다. man명령으로 확인해야한다.  
   -r 지정된 파일의 access time 및 modification time 으로 특정 파일의 시간을 변경한다.  
   [사용 예]  
   $ touch a.txt -> 현재 디렉터리에 a.txt라는 파일이 존재하면 파일의 수정시간을 바꾸고 파일이 없을 경우에는 a.txt라는 파일의 크기가 0인 빈 파일을 생성한다.  
   $ touch -r a.txt b.txt -> a.txt의 access time 및 modify time 으로 b.txt파일의 시간을 변경한다.**
7. **fie : 파일의 종류를 출력해주는 명령  
   [사용법] : $file [option] 파일명  
   -b : 정보 출력할 때 파일이름을 출력하지 않는다.  
   -I : 특정 파일의 문자 집합 정보를 출력한다.  
   [사용 예] $file -I lin.txt -> lin.txt 파일의 종류 및 문자집합 정보를 출력한다.**
8. **find : 현재 디렉터리부터 하위디렉터리까지 주어진 여러 조건을 가지고 파일 및 디렉터리를 찾아서 그 경로를 출력해주는 명령이다. root 에서 접근 권한을 제한한 파일이나 디렉터리에 관해서는 ‘허가거부됨’ 이라는 메시지가 뜨고 찾지를 못한다.  
   [사용법]  
   $find [path] options expression -> path 이하의 모든 디렉터리에서 파일을 찾는다. path가 없으면 현재디렉터리가 기준이다.  
   -name : 주어진 이름의 파일이나 디렉터리를 찾는다.  
   -user : 지정한 사용자의 파일이나 디렉터리를 찾는다.  
   -group : 지정한 그룹 소유의 파일이나 디렉터리를 찾는다.  
   -uid : 지정한 uid 소유의 파일이나 디렉터리를 찾는다.  
   -gid : 지정한 gid 소유의 파일이나 디렉터리를 찾는다.  
   -perm : 지정한 허가권 값을 갖는 파일이나 디렉터리를 찾는다.  
   -type : 지정한 파일 유형을 찾는다.  
   -atime n : n일 이전에 접근한파일을 찾는다.  
   -ctime n : change time 기준으로 찾는다.  
   -mtime n : modify time 기준으로 찾는다.  
   -exec : 찾아진 파일등과 연계하여 어떠한 명령을 내릴 수 있는 옵션이다.  
   -ok : 확인메시지를 출력하여 명령실행시 y나 n 을 입력받는 것을 제외하고는 -exec와 같다.  
   -print : 화면으로 출력을 받기위해서는 맨뒤에 꼭 써야하는 옵션이지만, 요즘 리눅스에서는 생략가능하다.  
   -fprint : 출력 결과를 파일로 저장할 때 사용하는 옵션이지만, ‘>’ 기호를 이용하면 된다.  
   -size n : n 블록 길이의 파일을 찾는다.  
   -empty : 파일의 크기가 0이거나 비어있는 디렉터리를 검색한다.  
   2>/dev/null : 옵션은 아니지만 맨 뒤에 이것을 사용하면 접근 권한이 없는 파일이나 디렉터리에 접근 시 나오는 ‘Permission Denied” 관련 리스트는 화면에 출력하지 않는다.  
   [사용 예]  
   $find =name ‘\*.c’ -> \*.c로 끝나는 파일이나 디렉터리를 찾는다.  
   #find -user posein ->posein 이라는 사용자의 파일이나 디렉터리를 찾는다.  
   $find -perm 600 ->퍼미션값이 600인 파일을 찾는다.  
   $find -type l ->심볼릭 링크인 파일을 찾는다.  
   $find . ->현재 디렉터리와 하위 디렉터리를 포함하여 모든 파일이나 디렉터리를 찾는다.  
   $find . -type d -> 현재 디렉터리에 있는 모든 하위 디렉터리를 찾는다.  
   $find . -type f -name “\*.txt” -> 현재 디렉터리에서 ‘\*.txt’로 끝나는 파일들을 찾는다.  
   $find / -type f -perm +6000 -ls -> 최상위 디렉터리에서부터 Set-UID와 Set-GID 둘 중 하나 이상이 설정된 파일들을 찾아 상세출력(ls)한다.  
   $find / -nouser -o -nogtoup -> 소유자나 소유그룹이 없는 파일들을 찾는다. -o는 or(또는)의 역할을 한다.  
   $find . -name ‘\*.txt’ -type f -exec rm {} \ ; ->현재 디렉터리에서 \*.txt로 된 파일들을 찾아 삭제한다.**