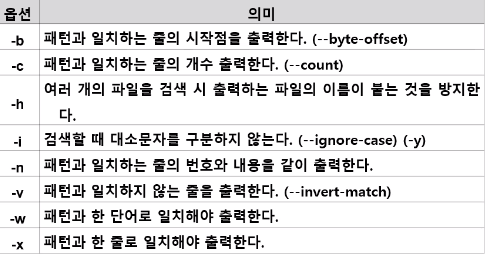
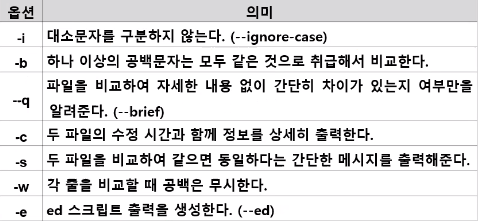
Report

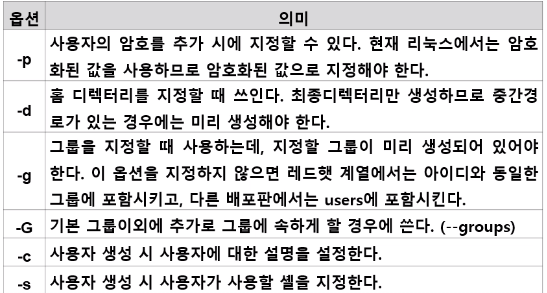
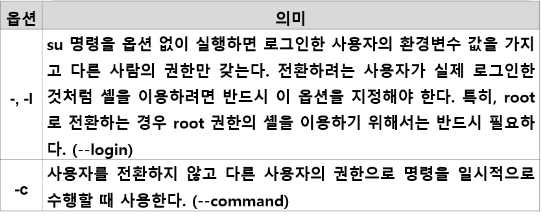
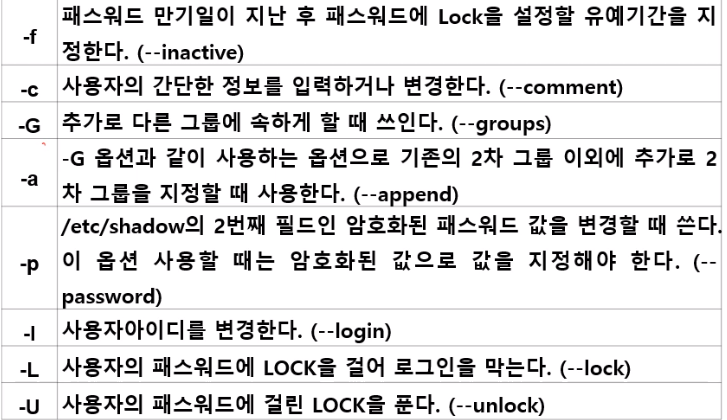
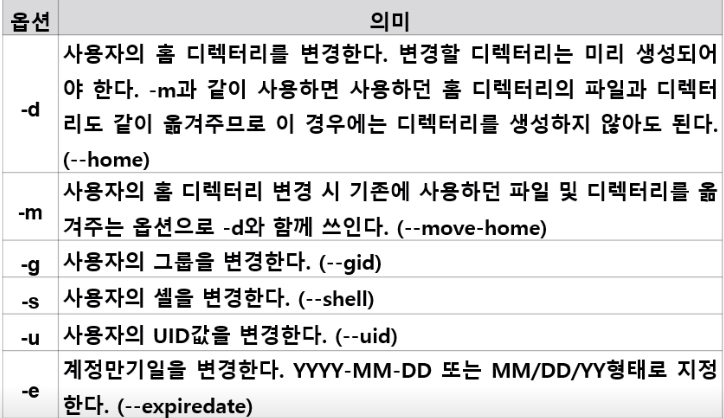
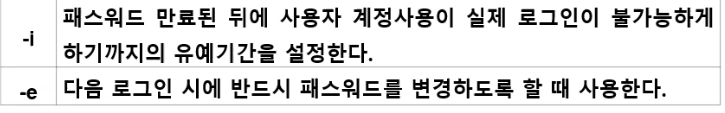
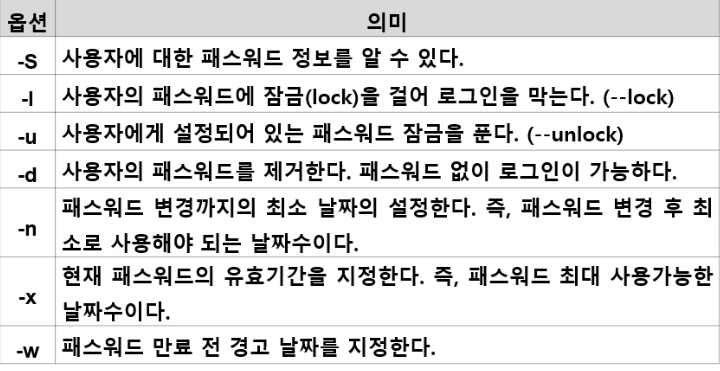
서버관리실무

**제목 : 5주차 과제**

**학과 : 컴퓨터소프트웨어과  
학번 : 2017100301  
이름 : 이상남  
 담당교수 : 김형철 교수님  
제출일 : 2020.09.29**

1. **cat(concatenate)  
   표준입력으로 받는 값을 표준출력으로 이어주는 명령으로 기본적으로는 텍스트 파일의 내용을 출력한다.  
   리다이렉션(redirection)기호와 함께 사용해서 파일을 생성하거나 여러 개의 텍스트 파일을 합치는 기능도한다.  
   -b 텍스트 파일 출력할 때 행번호를 붙여준다.(공백있는 줄 제외)  
   -n 텍스트 파일 출력할 때 행번호를 붙여준다.(공백있는 줄 포함)  
   리다이렉션은 입출력의 방향을 바꾸는 역할을 한다.  
   > : 출력전환(표준출력인 모니터에서 파일이나 기타 장치로 전환)  
   >> : 특수 출력( 보통 파일로 저장할 때 파일의 뒷부분에 추가되어 덧붙여짐)  
   <: 입력전환(표준입력인 키보드에서 파일이나 기타 장치로 전환)  
   << : 특수 입력(제시된 문자열이 입력될 때 까지 대기했다가 한 번에 표준출력으로 보낸다.)  
   [사용예]  
   $ cat -> 표준입력(키보드)을 표준출력(모니터)로 출력한다.  
   $ cat > a.txt -> 키보드로부터 입력받은내용을 a.txt라는 파일에 저장한다.  
   $ cat < a.txt -> a.txt 파일의 내용을 화면에 출력해준다.  
   $ cat <a.txt >b.txt -> a.txt의 내용을 b.txt라는 파일에 출력한다.   
   $cat a.txt b.txt -> a.txt내용과b.txt내용을 화면에 출력한다.  
   $ cat a.txt b.txt > c.txt -> a.txt,b.txt 파일의 내용을 합해서 c.txt라는 파일로 저장한다.  
   $ cat>>a.txt ->기존 내용을 유지한채로 내용을 추가한다. 파일이없을경우 cat>a.txt와 같다.  
   $cat << end -> end라는 문자열이 입력될때까지 기다렸따가 해당문자열이 입력되면 대기되었던 내용을 한 번에 표준 출력으로 보낸다.**
2. **head  
   텍스트 파일의 첫 부분을 보여주는 명령으로 옵션 없이 사용하면 기본 10줄을 출력한다.  
   -n 행수 : 파일의 앞부분부터 지정된 줄까지 출력한다.  
   [사용 예]  
   $head -3 \*.c -> .c로끝나는 모든파일의 처음 3줄씩을 보여준다.**
3. **tail  
   텍스트 파일의 끝 부분을 보여주는 명령으로 옵셔넚이 사용하면 기본 10줄을 출력한다.  
   [사용 예]  
   $tail/etc/passwd -> /etc/passwd의 마지막 10줄을 보여준다.**
4. **more  
   텍스트 파일의 내용이 긴 경우 화면 단위로 출력해주는 명령이다.  
   -num 한페이지를 num수만큼 출력한다.  
   space : 다음페이지  
   enter : 다음줄  
   q : 종료  
   [사용 예]  
   $more /etc/passwd ->/etc/passwd 파일의 내용을 한 페이지씩 출력한다.**
5. **less  
   more명령과 유사하게 한 화면단위로 출력해주는 명령으로 GNU에서 more명령의 성능을 강화한 뒤에 반대의 뜻을 붙여서 만들었다.  
   특히 ,커서(Cursor)키를 사용해서 상하좌우의 이동이 가능하고 vu에서 사용하는 다양한 명령들도 있다.**
6. **grep(Golbal Regular Expression Print)  
   텍스트파일에서 특정 패턴을 갖는 줄을 찾아서 출력해주는 명령이다.** **[사용 예]  
   $grep linux \* -> 현재 디렉터리의 모든파일에서 ‘linux’라는 문자열이 들어가 있는 줄을 찾아서 출력한다.  
   $grep “[0-9]” reademe -> reademe 파일에서 숫자로 시작하는 모든 라인을 보여준다.  
   정규표현식 : 쉽고 다양하고 정교하게 찾을 수 있도록 지원해주는 특정한 패턴을 말한다.  
   . : 매칭되는 한 문자로 무조건 존재해야 한다.  
   .\* : \*는 바로앞 한문자와 같이 행동하는데. 한 문자에 추가로 0이거나 그 이상의 문자를 나타낸다. 전부를 뜻한다  
   a\* : 아무것도 없거나 a 라는 문자열이 하나이거나 반복되어진 문자열을 뜻한다.  
   [abc] 는 문자리스트 중의 한 문자를 나타낸다. a,b,c를 의미한다.  
   [0-9] : 0~9Rkwlfmf dmlalgksek.**
7. **wc(word count)  
   텍스트파일의 행(line) 수, 단어(word) 수, 문자(byte) 수를 출력해주는 명령이다.  
   -ㅣ : 행 수만 출력한다.  
   -w : 단어 수만 출력한다.  
   -c : 문자 수만 출력한다.  
   [사용 예]   
   $wc readme -> readme 파일의 행, 단어, 문자의 수를 출력한다.  
   $who | we-l -> 현재 접속한 사람의 수를 표시한다.**
8. **sort  
   텍스트 파일의 내용을 행 단위로 정렬한다.  
   옵션을 지정하지 않으면 공백, 숫자, 특수문자(ASCII 코드순) , 대문자, 소문자 순이다.  
   [사용 예]  
   $sort jalin.txt -> jalin.txt를 공백, 숫자, 특수문자, 대문자, 소문자 순으로 행 단위 정렬하여 출력한다.**
9. **split  
   하나의 파일을 여러 개의 작은 파일로 분리하는 명령으로 옵션을 지정하지 않고, 실행하면 기본값이 1000줄 단위로 파일을 분리한다.  
   별도의 파일명을 지정하지 않으면 분리되어 생성되는 파일명이 xaa,xab,xac등으로 x뒤에 알파벳 순서로 생성된다.**
10. **diff(difference)  
    두 개의 파일을 행 단위로 비교하여 차이점을 출력해주는 명령이다.  
    보통 앞부분에 파일 원본을 놓고, 뒷 부분에 새로운 파일을 두고 비교한다.**
11. **사용자의 분류  
    리눅스의 사용자는 크게 root 사용자와 일반사용자로 구분한다.  
    root는 시스템운영에 있어서 모든 권한을 행사하므로 Privileged User 또는 Super User라고 한다.  
    일반 사용자는 보통 Normal user 또는 Unprivileged user 라 부르는데 시스템에 대해 제한적인 권한을 행사한다.  
    보통 root 라는 계정 자체를 슈퍼 유저라 인식하는데, 리눅스 시스템 내부에서는 사용자를 흔히 말하는 id로 관리하는 것이 아닌 숫자값 형태의 uid 로 관리한다. root는 0이 할당되고 0번사용자를 슈퍼유저로 인식한다.**
12. **시스템 계정  
    리눅스 사용자 계정의 정보는 /etc/passwd 파일에 기록되는데, 이 파일에 등록된 계쩡을 보면 root 이외에 bin,daemon,adm,game등 관리자가 생성하지 않는 계쩡들이 존재한다.  
    이러한 계정을 시스템 계정이라고 하는데, 말 그대로 시스템의 필요에 의해 생성된 계정이다.**
13. **일반 사용자를 슈퍼유저로 전환  
    일반사용자도 root처럼 UID와GID를 0 으로 바꾼다면 슈퍼유저로 전환되다.**

**사용자 생성 명령어**

1. **useradd  
   계정이란 시스템에ID와 암호를 생성하여 사용권을 부여하는 것을 말한다.  
   이 때 사용자의 ID를 생성하는 명령이 useradd이다.  
   root 권한자가 root 이외의 사용자를 생성할 때 사용하는 명령어로 adduser라고 입력해도 된다.** **[사용 예]  
   #useradd posein 0> posein이라는 계쩡을 생성한다.**
2. **passwd  
   등록된 사용자의 암호를 지정하거나 변경하는 명령어 이다.  
   리눅스에서는 useradd로 계쩡을 생성한 후에 암호를 지정하지 않으면 로그인이 되지 않으므로 반드시 지정한다.  
   root만 다른 사용자의 암호를 변경할 수 있다.**
3. **su  
   su(substitute user)는 대리 유저라는 뜻의 약어로 시스템에 로그인 되어 있는 상태에서 다른 사용자의 권한으로 셸을 실행할 수 있도록 전환하는 명령이다.** **[사용 예]  
   $ su -> root 사용자로 사용자를 전환한다. root로 권한자만 바꾸지만 환경변수 등은 기존사용자인 posein의 환경으로 적용되어 실제 root 권한을 수행하기에는 제약이 따른다.**
4. **사용자 관련 파일  
   리눅스는 대부분의 정보를 텍스트파일에 저장해서 관리한다.  
   사용자 계정 정보 역시 파일로 저장되고, 사용자를 생성할 때도 관련파일에서 정보를 가져온다.  
   useradd 명령을 실행하면 기본 설정은 /etc/default/useradd에서 정보를 가져오고, /ect/skel에 들어있는 파일 및 디렉터리를 사용자에게 제공한다.  
   생성된 사용자의 정보는 /etc/passwd와 /etc/shadw에 기록된다.**
5. **/etc/passwd  
   /etc/passwd는 시스템에 로그인하여 자원을 이용할 수 있는 사용자의 목록을 저장하고 있는 정보 파일이다.  
   콜론 (:)을 구분자로 ID 개인의 홈디렉터리 등 7개의 기본적인 정보를 담고 있으며, 패스워드를 다시 암호화 하여 /etc/shadow에서 추가로 관리한다.**
6. **/etc/shadow  
   /etc/passwd의 두 번째 필드인 패스워드 부분을 암호화하여 관리하는데 총 9개의 필드로 구성되어 있고, 패스워드 만기일 계정 만기일 등을 설정할 수 있다.  
   오직 root만 접근할 수 있다.**
7. **usermod  
   사용자의 셸, 홈 디렉터리, 그룹, UID, GID 등을 변경하는 명령어로 사용자 관련하여 대부분의 정보를 변경할 수 있다.**
8. **userdel  
   사용자 계정을 삭제하는 명령어 이다.  
   -r : 사용자의 홈 디렉터리 및 메일관련 파일까지 제거한다.**
9. **passwd  
   기본적으로 패스워드를 부여하거나 패스워드를 변경하는 명령이지만, 계정을 사용하지 못하게 하거나 패스워드 만기일 및 유효기간등을 설정할 수 있다.**
10. **chage  
    사용자의 패스워드에 대한 정보를 출력하고 설정하는 명령이다.**