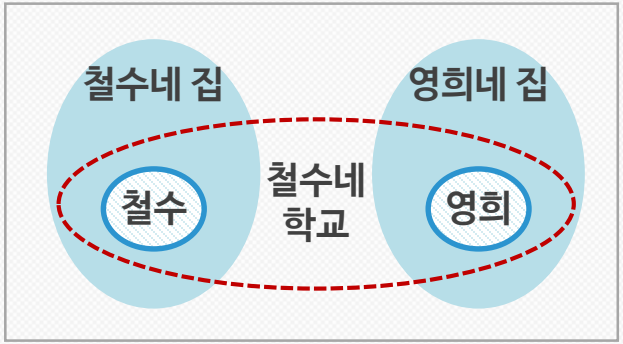
1. 사용자 및 그룹 관리

1) 사용자, 그룹, 권한 관리

"사용자와 그룹의 내용은 앞의 강의에서 언급하였던 부분으로
다음 도표를 기억해 봄"

✔ 유닉스,리눅스는 여러 사람이 사용하는 다중사용자 운영체계로
사용자 묶음의 그룹 개념이 존재

✔ 사용자는 여러 개의 그룹에 포함될 수 있음



- 철수라는 user는 [철수네 집], [철수네 학교] 그룹에 포함
- 영희는 [영희네 집], [철수네 학교] 그룹에 포함
- 철수는 [영희네 집] 그룹과는 "other" 관계(관계 없음)

1) 사용자, 그룹, 권한 관리

- ✔ 유닉스,리눅스는 여러 사람이 사용하는 다중사용자 운영체계로
  사용자 묶음의 그룹 개념이 존재

- ✔ 사용자는 여러 개의 그룹에 포함될 수 있음

구분	설명
[id], [groups]	현재의 사용자와 그룹을 알아보는 명령
[adduser], [addgrou]	사용자 등록, 그룹 등록
[deluser], [delgroup]	사용자 삭제, 그룹 삭제

1) 사용자, 그룹, 권한 관리

“ 사용자와 그룹은 시스템 내부에서 숫자로 표시됨 ”

uid
(사용자 표시 숫자)

gid
(그룹 표시 숫자)

1) 사용자, 그룹, 권한 관리

```
sjcu@sjcu:~$ id
uid=1000(sjcu) gid=1000(sjcu) groups=1000(sjcu),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),114(lpadmin),115(sambashare)
sjcu@sjcu:~$ groups
sjcu adm cdrom sudo dip plugdev lpadmin sambashare
sjcu@sjcu:~$
```

〈출처 : 교수자 제작물〉

- sjcu이라는 사용자는
  - uid(사용자id)가 1000
  - gid(그룹id)는 1000
  - 해당되는 그룹은 sjcu이라는 그룹 한 개에만 속함

2) 그룹 관리 명령어

1 그룹 조회

- 자기가 속한 그룹 : groups , id
- 전체 그룹을 보기 : /etc/group 파일을 봄

2 그룹 생성 : [groupadd]

예 groupadd -g 900 sjgroup : sjgroup라는 그룹을 gid를 900번으로 생성

2) 그룹 관리 명령어

3 그룹 변경 : [groupmod]

예
- groupmod -g 700 sjgroup : sjgroup라는 그룹을 gid를 700번으로 변경
- groupmod -n newsj sjgroup : sjgroup라는 그룹 명칭을 newsj로 변경

4 그룹 삭제 : [groupdel]

예 groupdel newsj : newsj라는 그룹을 삭제

3) 패스워드, 그룹 관련 설정 파일

/etc/passwd 파일

- 사용자의 정보(user, password, uid, pid)가 기록됨
- 해당 파일의 수정, 삭제 등으로 사용자 관련 설정 변경도 가능

/etc/group 파일

- 그룹의 정보가 기록됨
- 해당 파일의 수정 삭제 등으로 사용자 관련 설정 변경도 가능

3) 패스워드, 그룹 관련 설정 파일

/etc/shadows 파일

/etc/passwd 파일과 함께 사용자 패스워드를 저장

주의

- 단, 패스워드는 암호화 되어있는 문장으로 패스워드를 함부로 바꿀 수 없음
- 이 파일에서 패스워드 필드를 고치면 오류가 발생함

3) 패스워드, 그룹 관련 설정 파일

passwd 파일 구성 예

```
username:password:uid:gid:gecos:homedir:shell
```

구분	설명
username	사용자 명
password	사용자 암호
uid, gid	사용자 아이디, 그룹 아이디
gecos	▪ General Electric Comprehensive Operation System ▪ 예전 Unix 서비스와 호환성을 갖추기 위해 만든 필드 ▪ 처음 사용자 정보 넣은 값들이 저장
homedir	해당 사용자의 기본 디렉토리
shell	해당 사용자가 사용하는 Unix shell의 종류

3) 패스워드, 그룹 관련 설정 파일

passwd 파일 구성 예

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
syslog:x:101:103::/home/syslog:/bin/false
mysql:x:102:105:MySQL Server,,,:/nonexistent:/bin/false
messagebus:x:103:106::/var/run/dbus:/bin/false
"/etc/passwd" [readonly] 29L, 1415C
```

〈출처 : 교수자 제작물〉

2. 권한의 이해 및 표기

1) 파일이나 디렉토리의 소유자

> 처음 파일이나 디렉토리를 생성한 User의 소유로 생성

예

s1111111이라는 사용자로 접속하여 파일을 생성

➡ s1111111의 소유권

1) 파일이나 디렉토리의 소유자

[chown]

파일 또는 디렉토리의
소유 사용자를 바꿈

예 chown s1111111 aa

➡ aa파일을 s1111111 라는
사용자의 소유로 바꿈

gunzip

파일 또는 디렉토리의
소유 그룹을 바꿈

예 chgrp s1111111 aa

➡ aa파일을 s1111111 라는
그룹의 소유로 바꿈

2) 권한(permition)의 이해

유닉스 리눅스 3단계 권한

1. **읽기 권한** : 파일을 조회할 수 있음
2. **쓰기 권한** : 파일을 변경하거나 지울 수 있음
3. **실행 권한** : 파일을 실행할 수 있음

➡ 읽기 권한만 있으면 파일 조회는 가능하나 고치지는 못함

➡ 해당 파일이 실행파일이라면 실행하기 권한이 없으면 실행할 수 없음

✔ 권한 부여는 소유자(user), 그룹(group), 다른 사용자(other)에 대하여 부여함

2) 권한(permition)의 이해

✔ ls -al 명령으로 보여지는 내용으로 각각의 권한을 알 수 있음

```
sjcu@sjcu:~$ ls -al
total 44
drwxr-xr-x 4 sjcu sjcu 4096 Jan 18 11:16 .
drwxr-xr-x 4 root root 4096 Jan 18 11:09 ..
-rw------- 1 sjcu sjcu 577 Jan 18 11:11 .bash_history
-rw-r--r-- 1 sjcu sjcu 220 Dec 23 11:44 .bash_logout
-rw-r--r-- 1 sjcu sjcu 3486 Dec 23 11:44 .bashrc
drwx------ 2 sjcu sjcu 4096 Dec 23 11:45 .cache
-rw-r--r-- 1 root root 7 Jan 18 10:50 help2.txt
-rw-r--r-- 1 root root 7 Jan 18 10:50 hepl3.txt
drwxrwxr-x 2 sjcu sjcu 4096 Jan 18 11:06 mydir
-rw-r--r-- 1 sjcu sjcu 675 Dec 23 11:44 .profile
-rw------- 1 sjcu sjcu 3001 Jan 18 11:16 .viminfo
sjcu@sjcu:~$
```

〈출처 : 교수자 제작물〉

2) 권한(permition)의 이해

drwxr--r-- (d rwx r- r--)

ls에서 조회되는 처음 10개의 문자는 (1, 3, 3, 3)으로 나누어 보면

a) 처음 비트는 파일이면 - , 디렉토리면 d로 표시

b) 다음 3자리는 **소유자 (user)의 허가권**

c) 다음 3자리는 **그룹의 허가권**

d) 다음 3자리는 **소유자도 그룹도 아닌 자(other)의 허가권**

2) 권한(permition)의 이해

	구분	표시
1	파일 종류나 디렉토리임을 표시	d 또는 -
2	사용자의 읽기 권한	r 또는 -
3	사용자의 쓰기 권한	w 또는 -
4	사용자의 실행 권한	x 또는 -
5	그룹의 읽기 권한	r 또는 -

	구분	표시
6	그룹의 쓰기 권한	w 또는 -
7	그룹의 실행 권한	x 또는 -
8	다른 사람의 읽기 권한	r 또는 -
9	다른 사람의 쓰기 권한	w 또는 -
10	다른 사람의 실행 권한	x 또는 -

3) 권한(permition)의 숫자 표기법

파일 속성	소유자(user)			그룹(group)			다른 사람(other)		
	읽기 r(4)	쓰기 w(2)	실행 x(1)	읽기 r(4)	쓰기 w(2)	실행 x(1)	읽기 r(4)	쓰기 w(2)	실행 x(1)

- 사용자 r(4) w(2) x(1), 그룹r(4) w(2) x(1), 다른 사람 r(4) w(2) x(1)으로 표시한 값을 각 권한자 별로 더한 값으로 나타냄

예

사용자가 rx의 권한, 그룹이 wx의 권한, 다른 사람은 아무 권한을 가지지 않는다면

➡ 사용자 r(4) + x(1) = 5 , 그룹 w(2) + x(1) = 3, 다른 사람 0으로 보고 그 파일은 530의 권한을 가지고 있다고 표현

3. 권한 설정

1) 상대모드

[chomd]

- 권한 설정을 바꿈
- 바꾸는 방법에는 상대모드와 절대모드가 있음

〈상대모드 변경 방법〉

Operator	의미	Access class	의미
+	권한 부여	u	사용자
-	권한 제거	g	해당 그룹의 멤버들
=	권한 유지	o	다른 사람
S	사용자와 그룹만 실행	a	사용자, 그룹, 다른 사람 모두 권한 부여

1) 상대모드

예

- chmod g-w aaa : aaa파일에서 그룹의 쓰기권한을 제거
- chmod g+rw aaa : aaa파일에서 그룹의 읽기 쓰기권한 부여
- chmod a+x aaa : aaa파일에서 모두 실행권한을 부여
- chmod o-rwx aaa : aaa파일은 다른 사람은 읽거나, 쓰거나, 실행하지도 못하도록 함

2) 절대모드

"권한의 숫자 표기 방법으로 권한을 부여함"

파일 속성	소유자(user)			그룹(group)			다른 사람(other)		
	읽기 r(4)	쓰기 w(2)	실행 x(1)	읽기 r(4)	쓰기 w(2)	실행 x(1)	읽기 r(4)	쓰기 w(2)	실행 x(1)

2) 절대모드

예

사용자가 읽고,실행하고, 그룹 멤버는 쓰고 실행할 수 있는데, 다른 사람은 실행할 수 없게 하는 권한

- 사용자 r(4)+x(1) = 5, 그룹 w(2)+x(1)=3, 다른 사람 0이므로 권한의 숫자 표현은 530임
- 이런 권한을 aa파일에 부여한다면 [chmod 530 aa]로 명령

Tip

시스템 설정 파일 등은 관리자 이외에는 읽기, 쓰기, 실행하기 등을 제한하도록 함

4. 링크파일

1) 링크파일

링크파일

윈도우 바로가기 아이콘과 같이, 특정 파일이나 디렉토리를 연결해주는 역할의 빈 크기의 파일

2) Hard Link

- 하드링크의 두 파일명은 같은 디스크에 위치한 같은 데이터를 가리킴
- 하드링크는 원본 파일과 완전히 동일하고, 부가적인 디스크 공간을 차지하지 않음
- 하드링크 파일은 원본과 동일하기 때문에 하드링크 파일을 지우면 원본도 삭제됨
- 파일상태를 보는 명령은 stat

2) Hard Link

```
sjcu@sjcu:~$ stat abc
File: ‘abc’
Size: 7 Blocks: 8 IO Block: 4096 regular file
Device: fc00h/64512d Inode: 261898 Links: 2
Access: (0644/-rw-r--r--) Uid: ( 1000/ sjcu) Gid: ( 1000/ sjcu)
Access: 2016-01-18 11:17:24.532815359 +0900
Modify: 2016-01-18 11:17:24.532815359 +0900
Change: 2016-01-18 11:17:32.516815359 +0900
Birth: -
sjcu@sjcu:~$ stat l_abc
File: ‘l_abc’
Size: 7 Blocks: 8 IO Block: 4096 regular file
Device: fc00h/64512d Inode: 261898 Links: 2
Access: (0644/-rw-r--r--) Uid: ( 1000/ sjcu) Gid: ( 1000/ sjcu)
Access: 2016-01-18 11:17:24.532815359 +0900
Modify: 2016-01-18 11:17:24.532815359 +0900
Change: 2016-01-18 11:17:32.516815359 +0900
Birth: -
sjcu@sjcu:~$
```

- 좌측에서 ln abc l_abc로 하드링크를 실행한 경우, abc와 l_abc 는 동일한 파일이 연결되어 있음을 알 수 있음

〈출처 : 교수자 제작물〉

3) Symbolic Link

심볼릭 링크

- 작은 파일로 존재하고 이 파일은 링크된 파일
- 윈도우에서 바탕화면의 바로가기 아이콘 개념

“ 아이콘을 지운다고 해당 파일이 지워지지는 않음 ”

3) Symbolic Link

```
sjcu@sjcu:~$ ln -s efg l_efg
sjcu@sjcu:~$ stat efg
File: ‘efg’
Size: 7 Blocks: 8 IO Block: 4096 regular file
Device: fc00h/64512d Inode: 263076 Links: 1
Access: (0644/-rw-r--r--) Uid: ( 1000/ sjcu) Gid: ( 1000/ sjcu)
Access: 2016-01-18 11:18:17.332815359 +0900
Modify: 2016-01-18 11:18:17.332815359 +0900
Change: 2016-01-18 11:18:17.332815359 +0900
Birth: -
sjcu@sjcu:~$ stat l_efg
File: ‘l_efg’ -> ‘efg’
Size: 3 Blocks: 0 IO Block: 4096 symbolic link
Device: fc00h/64512d Inode: 263078 Links: 1
Access: (0777/lrwxrwxrwx) Uid: ( 1000/ sjcu) Gid: ( 1000/ sjcu)
Access: 2016-01-18 11:18:28.132815359 +0900
Modify: 2016-01-18 11:18:28.132815359 +0900
Change: 2016-01-18 11:18:28.132815359 +0900
Birth: -
sjcu@sjcu:~$
```

- 좌측에서 ln –s efg l_efg로 심볼릭 링크를 실행한 경우, efg와 l_efg 는 전혀 다른 파일임을 알 수 있음

〈출처 : 교수자 제작물〉

5. 사용자 그룹, 권한관리 실습

1) 실습하기

실습내용

(1) 사용자, 그룹 관리
(2) chown, chgrp 실습

6. 상대 · 절대모드, 링크 파일 실습

1) 실습하기

실습내용

(1) 상대모드 실습
(2) 절대모드 실습
(3) Hard Link
(4) Symbolic Link

학습활동

* 일시정지 버튼을 클릭하고 학습활동에 참여해 보세요.

Q 사용자와 권한관리의 이해와 권한관리 관련 명령어를 살펴 보았습니다.
이와 동일한 기능을 하는 windows의 명령어나 실행방법을 정리해 보세요.

※ 학습활동에 대한 해설

Q 사용자와 권한관리의 이해와 권한관리 관련 명령어를 살펴 보았습니다.
이와 동일한 기능을 하는 windows의 명령어나 실행방법을 정리해 보세요.

A 여러분은 파일과 디렉토리의 읽기, 쓰기, 실행하기에 대하여, 소유자, 그룹, 다른 사람에 대하여 권한을 부여하는 것을 배웠습니다.

이러한 내용은 유사하게 PC의 윈도우에서도 찾아볼 수 있습니다.

탐색기의 파일 속성에서는 파일을 읽기 전용으로 권한을 부여할 수 있습니다.
실행권한은 exe, com, bat과 같이 확장자에 따라 파일이 실행되고,
심지어 파일 확장자에 따라 연결 프로그램이 동작하여 해당 파일을 실행합니다.

어떤 실행파일은 시스템 권한이 부여되어야 실행되거나 읽을 수 있습니다.

드문 경우지만, 여러분이 윈도우에 다른 계정을 만들어 접속해 본다면 파일과 디렉토리에 다른 권한을 주어 접속하게 하는 것도 가능합니다.