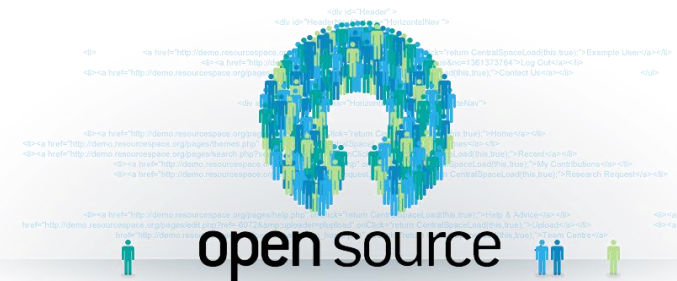




오픈소스 리눅스 실무

리눅스 기본 명령어3
파일





파일 다루기 (빈 파일 생성)

- touch
 - 내용이 없는 빈 파일을 생성한다.
 - 이미 파일이 있는 경우 수정 시간이 현재 시간으로 변경

touch [옵션] 파일

옵션 :

- a 접근 시간만 변경
- m 수정 시간만 변경
- t 시간을 직접 입력

- touch test

```
kdh@cs: ~/linux/ex
kdh@cs:~/linux/ex$ ls
kdh@cs:~/linux/ex$ touch test
kdh@cs:~/linux/ex$ ls -l
합 계 0
-rw-rw-r-- 1 kdh kdh 0 11월 19 10:24 test
kdh@cs:~/linux/ex$
```



파일 다루기 (파일 내용 검색)

- grep
 - 지정한 패턴이 포함된 행을 찾아 출력

grep [옵션] 패턴 파일

옵션 :

- i 대소문자 모두 검색
- l 지정한 패턴이 포함된 파일명을 출력
- n 행 번호를 출력



파일 다루기 (파일 내용 검색)

- grep -i 패턴 파일
 - 파일에서 패턴과 일치하는 대소문자 모두 검색

```
kdh@cs: ~/linux/ex
kdh@cs:~/linux/ex$ grep -i hello test
hello
Hello
HELLO
kdh@cs:~/linux/ex$
```



Linux

파일 다루기 (파일 내용 검색)

- grep -l 패턴 파일들
 - 파일들 중 패턴과 일치하는 내용을 가진 파일을 출력

```
kdh@cs: ~/linux/ex
kdh@cs:~/linux/ex$ grep -l hello *
exam
test
kdh@cs:~/linux/ex$
```



파일 다루기 (파일 내용 검색)

- grep -n 패턴 파일
 - 파일내용 중 패턴과 일치하는 내용의 라인을 포함해 출력

```
kdh@cs: ~/linux/ex
kdh@cs:~/linux/ex$ grep -n hello exam
1:"hello"
kdh@cs:~/linux/ex$ grep -n hello test
1:hello
kdh@cs:~/linux/ex$
```

- 두 개의 옵션 사용

```
kdh@cs: ~/linux/ex
kdh@cs:~/linux/ex$ grep -in hello test exam
test:1:hello
test:2:Hello
test:3:HELLO
exam:1:"hello"
kdh@cs:~/linux/ex$
```



파일 다루기 (파일 찾기)

- find
 - 지정한 위치에 조건에 맞는 파일 검색

find [옵션] [동작]

옵션 :

-name filename	파일명으로 검색
-type 파일종류	파일 종류로 검색
-user loginID	지정한 소유자가 소유한 파일
-perm 접근권한	지정한 권한과 일치하는 파일

동작 :

-exec 명령 { } \;	지정한 파일에 명령 실행
-ok 명령 { } \;	사용자의 확인을 받고 명령 실행
-print	지정한 파일의 절대경로 출력
-ls	검색 결과에 대한 목록을 출력



파일 다루기 (파일 찾기)

- find
 - find -name 파일명

```
kdh@cs: ~/linux
kdh@cs:~/linux$ find -name test
./ex/test
kdh@cs:~/linux$
```

- find -type 파일형식

```
kdh@cs: ~/linux
kdh@cs:~/linux$ find -type l 링크
./exam.softlink
kdh@cs:~/linux$ find -type d 디렉토리
.
./ex
kdh@cs:~/linux$
```




Linux

파일 다루기 (파일 찾기)

- find
 - find 디렉토리 -name 파일명
 - 검색 위치를 지정한 디렉토리에서 검색

```
kdh@cs: ~/linux
kdh@cs:~/linux$ find /bin -name ls
/bin/ls
kdh@cs:~/linux$
```



리눅스 파일

- 파일의 속성

파일 속성	의미
파일 크기	파일의 크기(K 바이트 단위)
파일 종류	일반 파일(-), 디렉터리(d), 링크(l), 파이프(p), 소켓(s), 디바이스(b 혹은 c) 등의 파일 종류를 나타낸다.
접근권한	파일에 대한 소유자, 그룹, 기타 사용자의 읽기(r)/쓰기(w)/실행(x) 권한
하드 링크 수	파일에 대한 하드 링크 개수
소유자 및 그룹	파일의 소유자 ID 및 소유자가 속한 그룹
파일 크기	파일의 크기(바이트 단위)
최종 수정 시간	파일을 생성 혹은 최후로 수정한 시간



리눅스 파일

- 파일의 속성
 - 파일 크기, 종류, 접근권한, 링크 수, 소유자 및 그룹, 수정 시간
- 파일의 속성 확인
 - `ls -sl exam`

```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -sl exam
4 -rw-rw-r-- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$
```

- 속성내용
 - 파일크기 4 kbyte , - : 일반파일
 - rw-rw-r-- :소유자 읽기 쓰기 , 그룹 읽기 쓰기 , 기타 읽기 권한
 - 하드링크 수 : 2 , 소유자 :kdh , 소유 그룹 : kdh , 파일 크기 : 13바이트
 - 최종수정일 : 11월 19 09:10 , 파일명 : exam



리눅스 파일

- 파일의 접근 권한
 - 리눅스는 파일에 대한 접근을 보호하는 기능을 제공
 - 사용자는 자신의 파일과 디렉토리에 다른 그룹이나 사용자가 접근해도 되는 것과 되지 않는 것을 구분하도록 접근 권한을 지정
- 파일의 접근 권한 확인
 - `ls -l`

```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -l
합 계 12
drwxrwxr-x 2 kdh kdh 4096 11월 19 10:37 ex
-rw-rw-r-- 2 kdh kdh 13 11월 19 09:10 exam
-rw-rw-r-- 2 kdh kdh 13 11월 19 09:10 exam.link
lrwxrwxrwx 1 kdh kdh 4 11월 19 09:26 exam.softlink ->
exam
kdh@cs:~/linux$
```



리눅스 파일

- 리눅스에서 지원하는 파일 종류

파일 종류	표시	설명
일반 파일	-	데이터를 갖고 있는 텍스트 파일 또는 이진 파일
디렉터리 파일	d	디렉터리 내의 파일들의 이름들과 파일 정보를 관리하는 파일
문자 장치 파일	c	문자 단위로 데이터를 전송하는 장치를 나타내는 파일
블록 장치 파일	b	블록 단위로 데이터를 전송하는 장치를 나타내는 파일
FIFO 파일	p	프로세스 간 통신에 사용되는 이름 있는 파이프
소켓	s	네트워크를 통한 프로세스 간 통신에 사용되는 파일
심벌릭 링크	l	다른 파일을 가리키는 포인터와 같은 역할을 하는 파일



리눅스 파일

- 파일 종류 확인 명령
 - File 파일명
 - 파일에 대한 자세한 정보를 출력

```
kdh@cs: ~/linux
kdh@cs:~/linux$ file exam
exam: ASCII text
kdh@cs:~/linux$
```

exam 파일은 아스키코드 텍스트 파일임을 출력



리눅스 파일

- 파일의 접근 권한
 - 접근 권한의 종류
 - 읽기권한 , 쓰기권한 , 실행권한

권한	파일	디렉터리
읽기	파일을 읽거나 복사할 수 있다.	ls 명령으로 디렉터리 목록을 볼 수 있다(ls 명령의 옵션은 실행 권한이 있어야 사용할 수 있다).
쓰기	파일을 수정·이동·삭제할 수 있다(디렉터리에 쓰기 권한이 있어야 한다).	파일을 생성하거나 삭제할 수 있다.
실행	파일을 실행할 수 있다(셸 스크립트나 실행 파일의 경우).	cd 명령을 사용할 수 있다. 파일을 디렉터리로 이동하거나 복사할 수 있다.



리눅스 파일

- 파일의 접근 권한

- 기본 접근 권한

- 일반파일 – 소유자는 읽기와 쓰기 그룹과 사용자는 읽기 권한만 설정
 - 디렉토리 – 소유자는 읽기,쓰기,실행 그룹과 사용자는 읽기와 실행 권한만 설정

```
kdh@cs:~/linux$ touch test.txt  
kdh@cs:~/linux$ mkdir test
```

```
drwxrwxr-x 2 kdh kdh 4096 11월 19 18:21 test  
-rw-rw-r-- 1 kdh kdh 0 11월 19 18:21 test.txt
```




리눅스 파일

- 파일의 접근 권한

- 접근 권한 표기 방법

- 읽기권한 , 쓰기권한 , 실행권한을 각각 r , w , x 로 표기
 - 소유자 , 그룹 , 기타 사용자별로 r,w,x 가 표기 (- 는 권한 없음을 의미)
 - 예 : `rw-r--r--`



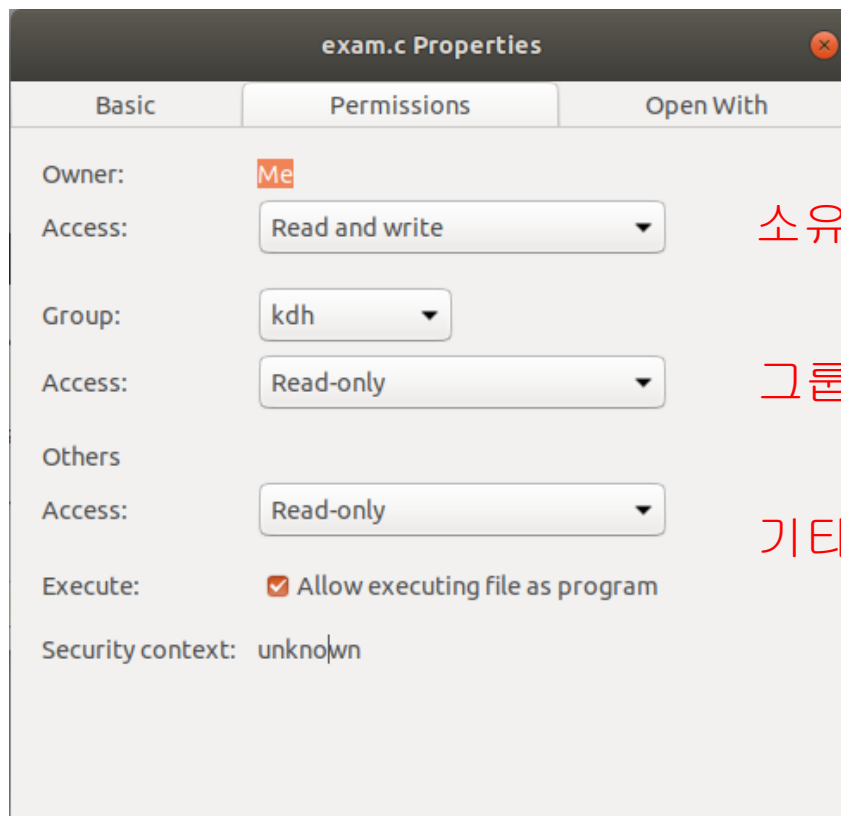
```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -l exam
-rw-rw-r-- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$
```

소유자는 읽기 쓰기 가능 , 동일그룹은 읽기 쓰기 가능 ,기타 사용자는 읽기 가능



리눅스 파일

- 파일의 접근 권한
 - GUI 환경에서의 속성확인





리눅스 파일

- 다양한 접근 권한의 예

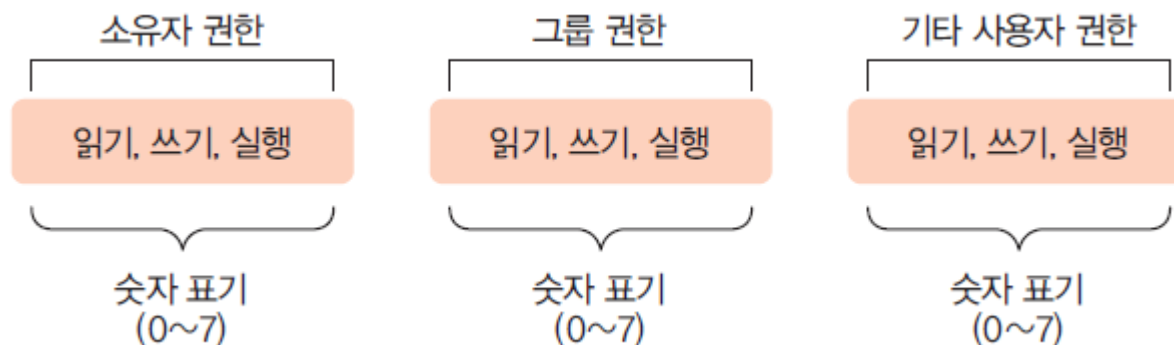
접근 권한	의미
<code>rw-r--r--</code>	소유자는 읽기·쓰기·실행 권한을 모두 가지고 그룹과 기타 사용자는 읽기·실행 권한을 가지고 있다.
<code>r--r--r--</code>	소유자, 그룹, 기타 사용자 모두 읽기·실행 권한을 가지고 있다.
<code>rw-----</code>	소유자만 읽기·쓰기 권한을 가지고 그룹과 기타 사용자는 아무 권한이 없다.
<code>rw-rw-rw-</code>	소유자, 그룹, 기타 사용자 모두 읽기·쓰기 권한을 가지고 있다.
<code>rw-rw-rwx</code>	소유자, 그룹, 기타 사용자 모두 읽기·쓰기·실행 권한을 가지고 있다.
<code>rw-----</code>	소유자만 읽기·쓰기·실행 권한을 가지고 그룹과 기타 사용자는 아무 권한이 없다.
<code>r-----</code>	소유자만 읽기 권한을 가지고 있다.



리눅스 파일

- 접근 권한의 변경

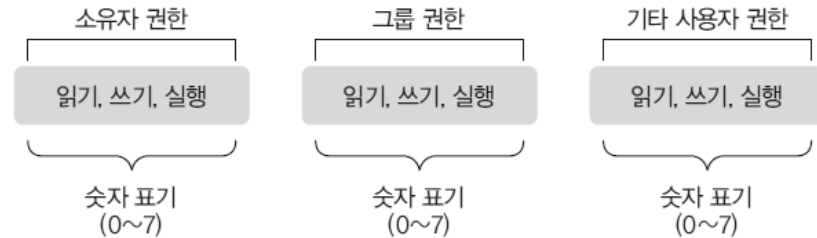
chmod [옵션] 접근권한 파일(디렉토리)
옵션 :
-R : 하위 디렉토리 포함



리눅스 파일



- 접근 권한의 변경
 - 숫자 모드 접근 권한



접근 권한	환산	숫자	의미
rwX	111 → 4+2+1	7	읽기, 쓰기, 실행
rw-	110 → 4+2+0	6	읽기, 쓰기
r-X	101 → 4+0+1	5	읽기, 실행
r--	100 → 4+0+0	4	읽기
-wX	011 → 0+2+1	3	쓰기, 실행
-w-	010 → 0+2+0	2	쓰기
--X	001 → 0+0+1	1	실행
---	000 → 0+0+0	0	권한이 없음

접근 권한	숫자 모드	접근 권한	숫자 모드
rwXrwXrwX	777	rw-r--r--	644
rwXr-Xr-X	755	rwX-----	700
rw-rw-rw	666	rw-r-----	640
r-Xr-Xr-X	555	r-----	400

- 접근 권한 변경 예
 - `chmod 755 exam`



리눅스 파일

- 접근 권한의 변경
 - 접근 권한 변경 예
 - `chmod 744 exam`

```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -l exam
-rw-rw-r-- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$ chmod 644 exam
kdh@cs:~/linux$ ls -l exam
-rw-r--r-- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$
```



리눅스 파일

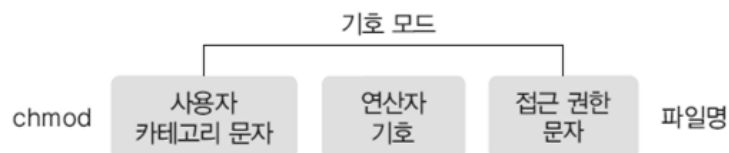
- 접근 권한의 변경
 - 접근 권한 변경 예
 - `chmod 600 exam`

```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -l exam
-rwxr--r-- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$ chmod 600 exam
kdh@cs:~/linux$ ls -l exam
-rw----- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$
```



리눅스 파일

- 접근 권한의 변경
 - 기호 모드 접근 권한



구분	문자/기호	의미
사용자 카테고리 문자	u	파일 소유자
	g	소유자가 속한 그룹
	o	소유자와 그룹 이외의 기타 사용자
	a	전체 사용자
연산자 기호	+	권한 부여
	-	권한 제거
	=	접근 권한 설정
접근 권한 문자	r	읽기 권한
	w	쓰기 권한
	x	실행 권한

- 예) `chmod g+w exam`
 - exam 파일을 같은 그룹에 대해 쓰기 권한을 부여



리눅스 파일

- 접근 권한의 변경
 - 기호 모드 접근 권한 설정의 다양한 예

권한 표기	의미
u+w	소유자(u)에게 쓰기(w) 권한 부여(+)
u-x	소유자(u)의 실행(x) 권한 제거(-)
g+w	그룹(g)에 쓰기(w) 권한 부여(+)
o-r	기타 사용자(o)의 읽기(r) 권한 제거(-)
g+wx	그룹(g)에 쓰기(w)와 실행(x) 권한 부여(+)
+wx	모든 사용자에게 umask에 따라 권한 부여(+)
a+rw	모든 사용자에게 읽기(r), 쓰기(w), 실행(x) 권한 부여(+)
u-rwx	소유자(u)에게 읽기(r), 쓰기(w), 실행(x) 권한 부여(=)
go+w	그룹(g)과 기타 사용자(o)에게 쓰기(w) 권한 부여(+)
u+x,go+w	소유자(u)에게 실행(x) 권한을 부여하고(+) 그룹(g)과 기타 사용자(o)에게 쓰기(w) 권한 부여(+)



리눅스 파일

- 소유자 변경

- 파일 혹은 디렉토리의 소유자를 지정된 사용자로 변경
- -R 옵션이 지정된 경우 하위 디렉토리의 소유자도 변경

`chown [-R] 사용자 파일(디렉토리)`

옵션 :

`-R` : 디렉토리일 경우 하위 디렉토리 포함



리눅스 파일

- 소유자 변경

- 소유자 변경의 경우 root 권한이 필요
- sudo 명령 : 해당 명령을 root 권한으로 실행

```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -l exam
-rw----- 2 kdh kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$ sudo chown root exam
[sudo] kdh의 암호 :
kdh@cs:~/linux$ ls -l exam
-rw----- 2 root kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$
```

■ 소유자가 변경



리눅스 파일

- 소유 그룹 변경

- 소유 그룹 변경의 경우 root 권한이 필요
- sudo 명령 : 해당 명령을 root 권한으로 실행

```
kdh@cs: ~/linux
kdh@cs:~/linux$ ls -l exam
-rw----- 2 root kdh 13 11월 19 09:10 exam
kdh@cs:~/linux$ sudo chgrp root exam
kdh@cs:~/linux$ ls -l exam
-rw----- 2 root root 13 11월 19 09:10 exam
kdh@cs:~/linux$
```

그룹이 변경