

파일 시스템 II

- 파일의 접근권한

컴퓨터소프트웨어학과

김병국 교수



- 파일의 접근권한을 읽을 수 있다.
- 접근 권한을 수정할 수 있다.
- 파일의 접근 권한을 변경하는 프로그램을 작성할 수 있다.



- 접근 권한 보기
- 접근 권한 수정
- 프로그램 함수
- 프로그래밍 실습



1. 접근 권한 보기 (1/4)

□ 파일 및 디렉토리 보기 (1/2)

■ 명령어: ls

- 지정한 경로(생략: 현재 위치)에 대한 파일들의 속성(permission, ownership, size, file name ...)을 출력
- 대표적 옵션:
 - -a(--all) : 숨겨진 파일과 디렉토리까지 출력
 - -l : (long) : 파일과 디렉토리에 대한 상세정보를 출력
 - -i : 파일의 i-node의 값을 출력
 - -R : 하위 폴더(sub-directories) 내 모든 파일들을 출력

```
[kali@kali:~]$ ls
Desktop  Downloads  NetworkProgramming  Pictures  Templates  Videos
Documents  Music      OperatingSystem      Public    test.c
[kali@kali:~]$ ls -l
total 44
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Desktop
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Documents
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Downloads
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Music
drwxr-xr-x 2 kali kali 4096 Mar 26 01:03 NetworkProgramming
drwxr-xr-x 7 kali kali 4096 Mar 29 23:09 OperatingSystem
drwxr-xr-x 2 kali kali 4096 Mar 30 08:37 Pictures
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Public
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Templates
-rw-r--r-- 1 kali kali  55 Mar 18 10:19 test.c
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Videos
[kali@kali:~]$
```

【명령어 실행 예】



1. 접근 권한 보기 (2/4)

□ 파일 및 디렉토리 보기 (2/2)

▪ 파일 속성

```
[kali@kali:~]$ ls -l
total 64
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Desktop
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Documents
```



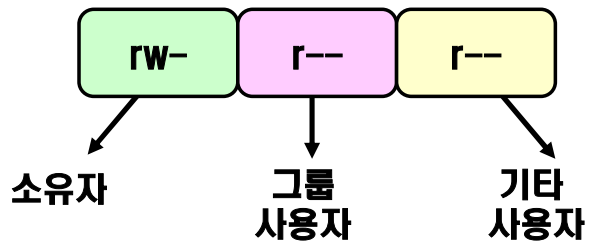
번호	값	동작
1	d	파일 종류 (- : 일반파일, d: 디렉토리)
2	rw-xr--r-x	파일을 읽고, 쓰고, 실행할 수 있는 접근 권한 표시
3	kali	파일 소유자의 사용자 명
4	kali	파일 소유자의 그룹명



1. 접근 권한 보기 (3/4)

□파일의 접근 권한 보기

```
[kali@kali:~]$ ls -l
total 64
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Desktop
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Documents
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Downloads
-rwxr-xr-x 1 kali kali 16608 Apr  2 01:37 hello
-rw-r--r-- 1 kali kali  72 Apr  2 01:37 hello.c
drwxr-xr-x 2 kali kali 4096 Feb 23 05:37 Music
drwxr-xr-x 2 kali kali 4096 Mar 26 01:03 NetworkPro
```

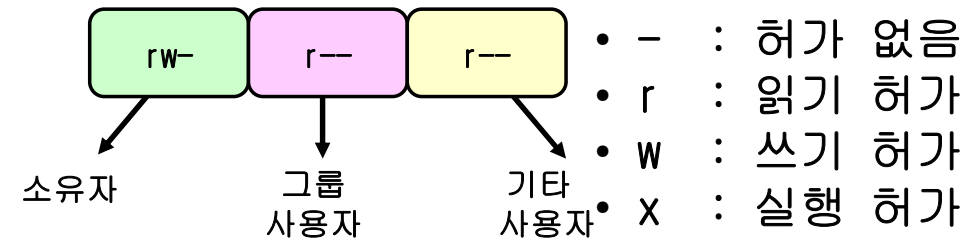


- - : 허가 없음
- r : 읽기 허가
- w : 쓰기 허가
- x : 실행 허가



1. 접근 권한 보기 (4/4)

□파일의 접근 권한 조합



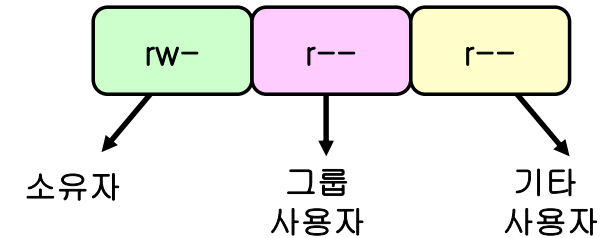
사용 권한	의미
<code>rwxr-xr-x</code>	소유자는 읽기/쓰기/실행 권한을 모두 가짐 그룹과 기타사용자는 읽기와 실행권한만 가짐
<code>r-xr-xr-x</code>	소유자, 그룹, 기타사용자 모두 읽기와 실행권한만 가짐
<code>rw-----</code>	소유자만 읽기/쓰기 권한을 가짐 그룹과 기타사용자는 아무 권한도 없음
<code>rw-rw-rw-</code>	소유자와 그룹, 기타사용자 모두 읽기와 쓰기 권한을 가짐
<code>rwxrwxrwx</code>	소유자, 그룹, 기타사용자 모두 읽기/쓰기/실행 권한을 가짐
<code>rwx-----</code>	소유자만 읽기/쓰기/실행권한을 가짐 나머지는 아무 권한도 없음



2. 접근 권한 수정 (1/3)

□ 파일의 접근 권한 수정 (1/2)

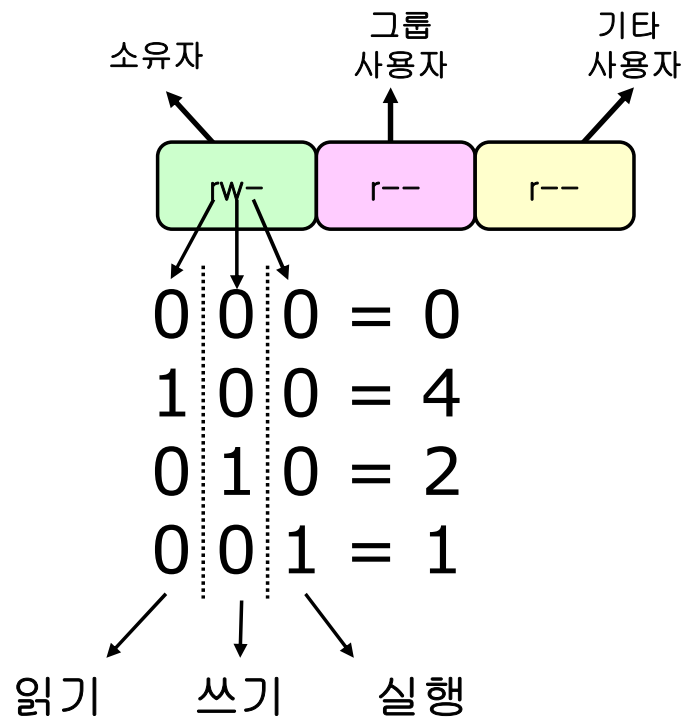
- 명령어: `chmod [모드] [대상 파일 또는 디렉토리]`
- 자신이 소유한 파일에 대해서만 접근 권한을 변경할 수 있음
 - 단, 시스템관리자(root)는 예외
- 권한 표시 방식
 - 권한 모드
 - 3자리의 영역을 8진수로 표현하는 방식
 - 예: `chmod 644 test.txt`
 - 기호 모드
 - 각 의미를 갖는 심벌(r, w, x)을 조합해서 적용
 - 예: `chmod a=rwx test.txt`



2. 접근 권한 수정 (2/3)

□ 권한 형태 : 권한 모드

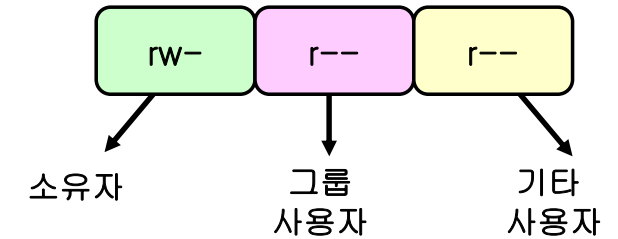
- 세 자리의 8진수 형태로 권한을 변경하는 방식



2. 접근 권한 수정 (3/3)

□ 권한 형태 : 기호 모드

- 기호 패턴을 이용하여 접근 권한을 변경



사용자 카테고리	의미	연산자 기호	의미
u	소유자	+	허가권 부여
g	그룹	-	허가권 제거
o	기타사용자	=	특정 사용자에게 허가권 지정
a	모든 사용자(u+g+o)	,	추가 연산

권한 기호	의미
r	읽기 허가
w	쓰기 허가
x	실행 허가



3. 프로그램 함수 (1/5)

□ 파일의 접근 권한 변경

- 함수: chmod()
 - 지정한 파일명에 대하여 접근권한을 변경
- 인자:
 - *pathname : 변경할 파일(또는 디렉토리경로)의 이름
 - mode : 권한(8진수 세자리 숫자)
- 결과 값:
 - 성공: 0
 - 실패: -1

```
#include <sys/stat.h>

int chmod(const char *pathname, mode_t mode);
int fchmod(int fd, mode_t mode);
```

[chmod() 함수의 프로토타입]



3. 프로그램 함수 (2/5)

실습

- chmod() 함수 이용

```
1  #include <stdio.h>
2  #include <sys/stat.h>
3  #include <unistd.h>
4
5  int
6  main(int argc, char* argv[])
7  {
8      int nMode = 0;
9
10     if (argc != 3)
11     {
12         printf("Usage: %s mode filename\n", argv[0]);
13         return -1;
14     }
15
16     sscanf(argv[1], "%o", &nMode);
17
18     printf("Filename : %s\n", argv[2]);
19     printf("Permission: 0%03o\n", nMode);
```

```
20
21     if (chmod(argv[2], nMode) == 0) {
22         printf("Success.\n");
23     } else {
24         printf("Failed.\n");
25     }
26
27     return 0;
28 }
```

[파일명: chmod.c]



3. 프로그램 함수 (3/5)

□ 파일의 접근 권한 변경

- 함수: fchmod()
 - 지정한 파일에 대하여 접근권한을 변경
 - 파일 기술자를 사용
- 인자:
 - fd : 파일 기술자
 - mode : 권한(8진수 세자리 숫자)
- 결과 값:
 - 성공 : 0
 - 실패 : -1

```
#include <sys/stat.h>

int chmod(const char *pathname, mode_t mode);
int fchmod(int fd, mode_t mode);
```

[fchmod() 함수의 프로토타입]



3. 프로그램 함수 (4/5)

□ 실습 1

- fchmod() 함수 이용

```
1  #include <stdio.h>
2  #include <sys/types.h>
3  #include <sys/stat.h>
4  #include <fcntl.h>
5  #include <unistd.h>
6
7  int
8  main(int argc, char* argv[])
9  {
10     int nFd = -1;
11     int nMode = 0;
12
13     if (argc != 3) {
14         printf("Usage: %s mode filename\n", argv[0]);
15         return -1;
16     }
17
18     nFd = open(argv[2], O_RDONLY);
19
```

```
20     sscanf(argv[1], "%o", &nMode);
21
22     printf("Filename : %s\n", argv[2]);
23     printf("Permission: 0%03o\n", nMode);
24
25     if (fchmod(nFd, nMode) == 0) {
26         printf("Success.\n");
27     } else {
28         printf("Failed.\n");
29     }
30
31     close(nFd);
32
33     return 0;
34 }
```

[파일명: fchmod.c]

3. 프로그램 함수 (5/5)

□ 실습 2

■ fchmod() 함수 이용

```
1  #include <stdio.h>
2  #include <sys/types.h>
3  #include <sys/stat.h>
4  #include <fcntl.h>
5  #include <unistd.h>
6
7  int main(int argc, char *argv[])
8  {
9      int nFd = -1;
10     int nLen = 0;
11
12     if (argc != 2)
13     {
14         printf("Usage: %s filename\n", argv[0]);
15         return -1;
16     }
17
18     nFd = open(argv[1], O_WRONLY | O_CREAT, 0644);
19
```

1

```
20     nLen = write(nFd, "Hello", 5);
21     if (nLen <= 0)
22     {
23         printf("Writting was failed(1).\n");
24     }
25
26     fchmod(nFd, 0444);
27
28     printf("Input Enter key to continue.\n");
29     getchar();
30
31     nLen = write(nFd, " World\n", 7);
32     if (nLen <= 0)
33     {
34         printf("Writting was failed(2).\n");
35     }
36
37     close(nFd);
38
39     return 0;
40 }
```

2

[파일명: fchmod2.c]

수고하셨습니다.

