상명대학교 컴퓨터과학부 "EA0014: 유닉스프로그래밍" 실습 지침서

- 실습 번호 : lab-03

- 실습 디렉토리 : ~/unix/lab-03 이름 : 임현

- 실습 날짜 : 2017년 04월 04일

- 실습 제목 : File & Directory

- 실습 내용 : 학번 : 201511054

1. (stat() syscall, 파일 “3-1.c”) $cp /etc/passwd password.txt 를 통해 디렉토리에 password.txt 파 일을 만든다.

아래와 같이 3-1.c를 편집하여 만들고



gcc를 이용 다음과 같이 컴파일 하시오.

$gcc –o 3-1 3-1.c

$./3-1 로 실행하고 결과를 적으시오.

$ls –lai 하여 위의 결과와 비교 설명하시오.



./3-1의 실행 결과는 위와 같습니다.

ls -lai와 ./3-1의 실행 결과를 비교 설명 하자면

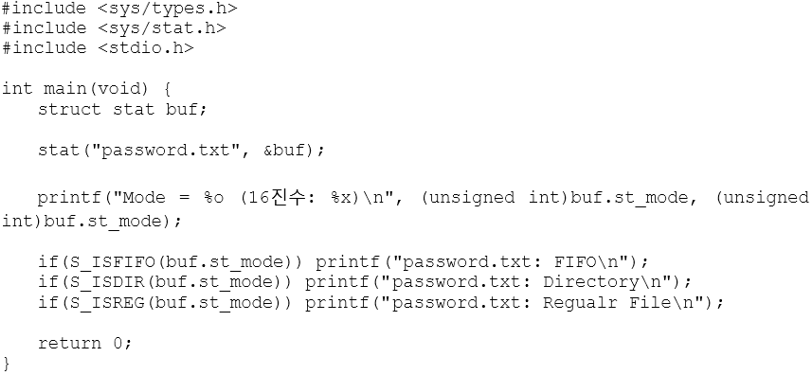
|  |  |  |  |
| --- | --- | --- | --- |
|  | ./3-1 | la -lai | 표현 방식 |
| Inode | 7996137 | 7996317 | 같음 |
| Mode | 100644 | -rw-r--r-- | 다름 |
| Nlink | 1 | 1 | 같음 |
| UID | 1063 | unix201511054 | 다름 |
| GID | 1051 | unixclass1 | 다름 |
| SIZE | 22344 | 22344 | 같음 |
| Atime | 1491271766 | 2017-04-04 11 : 09 | 다름 |
| Mtime | 1491271766 | 2017-04-04 11 : 09 | 다름 |
| Ctime | 1491271766 | 2017-04-04 11 : 09 | 다름 |
| Blksize | 4096 |  |  |
| Blocks | 48 |  |  |
| 파일명 | password.txt | password.txt | 같음 |

Mode는 100은 일반 파일, 644 및 ‘rw-r--r--‘는 접근 권한을 나타냄.

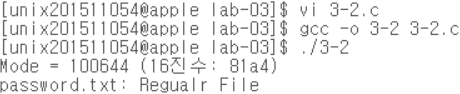
UID, GID는 표현 방식만 다르다고 볼 수 있음.

Time 관련 항목은 1970년 1월 1일 0시 0분 0초부터의 시간을 초로 환산해서 표현방식이 다름.

2. (파일종류 매크로 사용, 파일 “3-2.c”) 아래와 같이 “3-2.c”를 편집하여 만든다.



$ gcc –o 3-2 3-2.c 로 컴파일하고 $ ./3-2 를 통해 프로그램을 수행하고 결과를 적으시오.



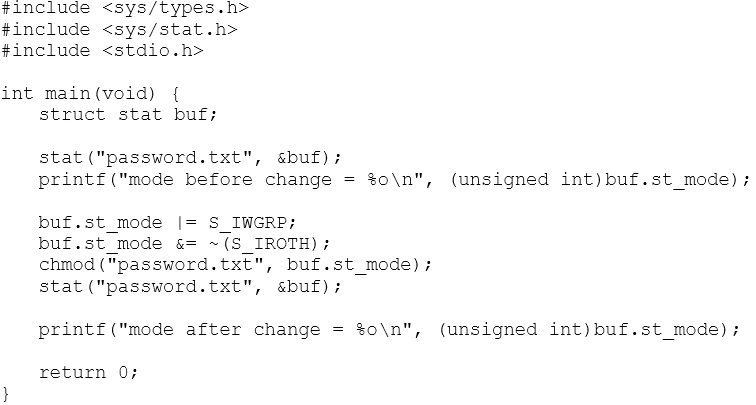
./3-2 의 수행 결과는 위와 같습니다.

접근 권한은 100644 (16진수 : 0x81a4)이므로, S\_ISREG(상수값 : 0x80000)과 일치합니다.

그러므로 ‘Regular File’ 이라는 결론이 나옵니다.

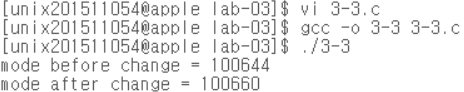
3. (chmod() syscall, 파일 “3-3.c”) ls –l로 password.txt의 접근 권한을 확인하시오.

아래와 같이 3-3.c를 만들고



$ gcc –o 3-3 3-3.c 를 통해 컴파일한후 $./3-3 으로 수행하고 결과를 적으시오.

다시 ls -l하여 처음과 비교하시오.

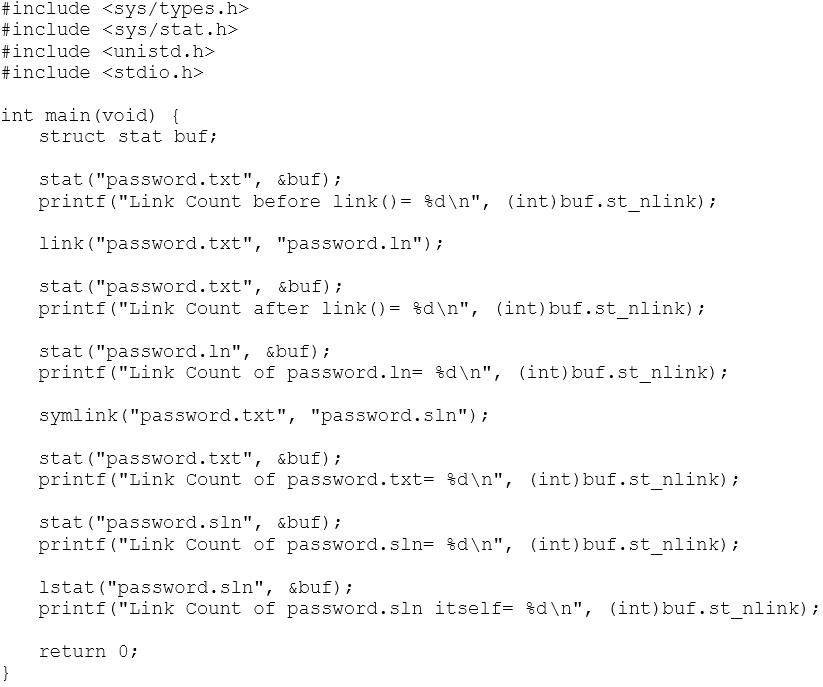


./3-3의 수행 결과는 위와 같습니다.

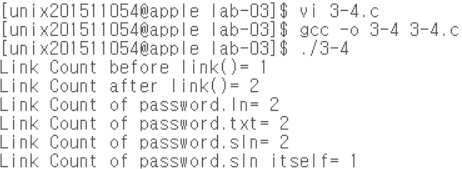
“password.txt” 파일의 접근 권한이 ‘644’에서 ‘660’으로 바뀌면서,

그룹 접근 권한이 r에서 rw로, 기타 사용자 접근 권한이 rw에서 접근 불가로 바뀌었습니다.

4. (link(), symlink() syscall, 파일“3-4.c”) 아래 프로그램과 같이 “3-4.c”를 만들고 컴파일 후 수행하 여 결과를 확인하고 왜 그런 결과가 나오는지를 설명하시오.



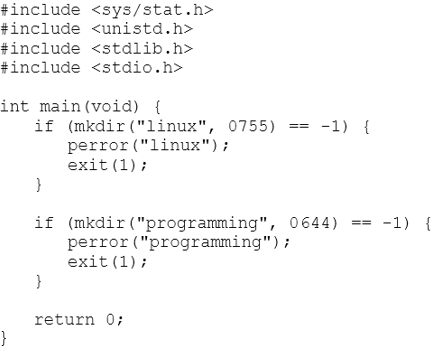
./3-4의 수행 결과는 아래와 같습니다.



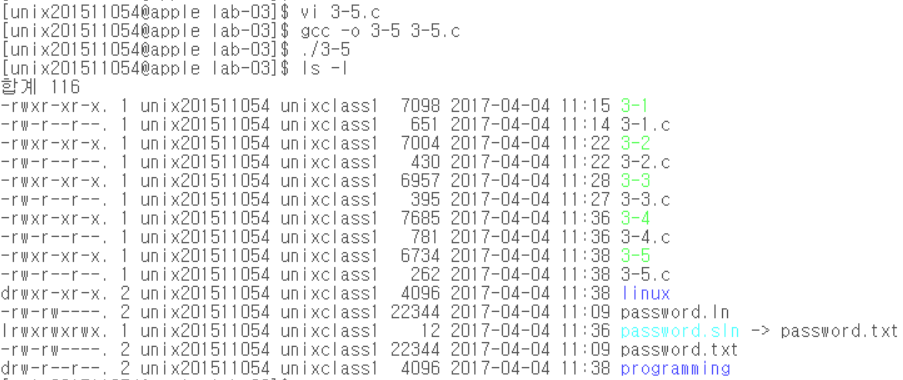
password.txt 파일에 하드 링크와 심볼릭 링크를 생성해주었습니다.

after link가 2개가 나오는 이유는 심볼릭 링크의 경우 다른 inode를 쓰면서, 기존 파일의 경로만 제공해주기 때문에 원래 파일과 하드링크 총 2개만 나오는 것 입니다.

5. (mkdir() syscall, 파일 “3-5.c”) linux, programming이란 이름의 디렉토리를 만드는 프로그램 35.c를 작성하고 컴파일, 수행하시오.



$ ls –l을 통해 디렉토리가 만들어 진 것을 확인하시오.

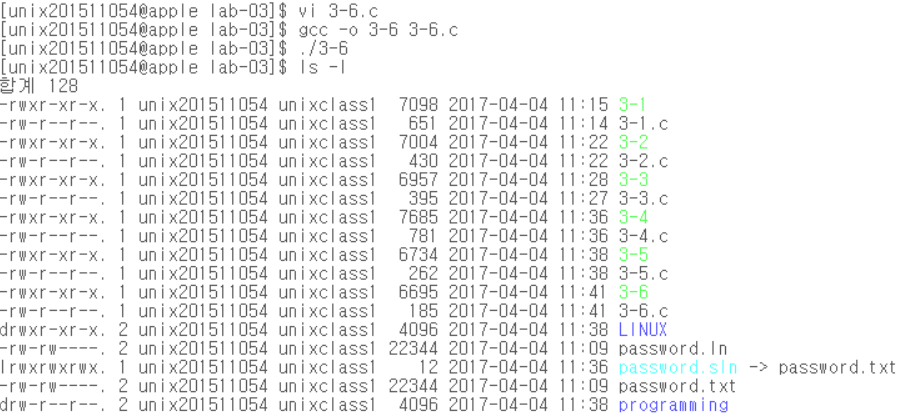


‘rwxr-xr-x’ 접근 권한을 가지고 있는 ‘linux’ 폴더와

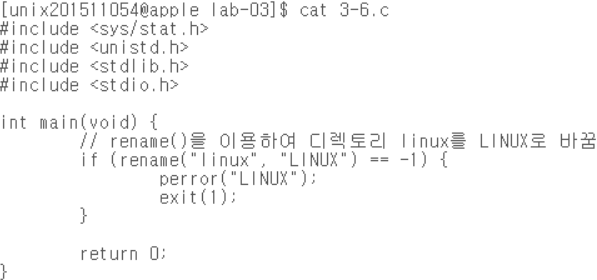
‘rw-r--r--‘ 접근 권한을 가지고 있는 ‘programming’ 폴더가

만들어졌습니다.

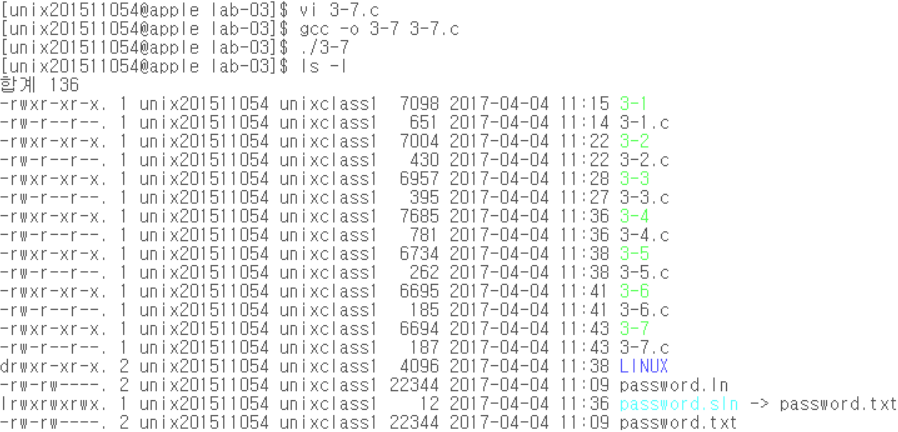
6. (rename() syscall, 파일 “3-6.c”) 시스템 콜 rename()을 이용하여 디렉토리 linux를 LINUX로 바꾸 는 프로그램 3-6.c 를 작성하고 컴파일, 수행하시오. $ ls –l로 디렉토리의 이름이 바뀐 것을 확인 하시오.



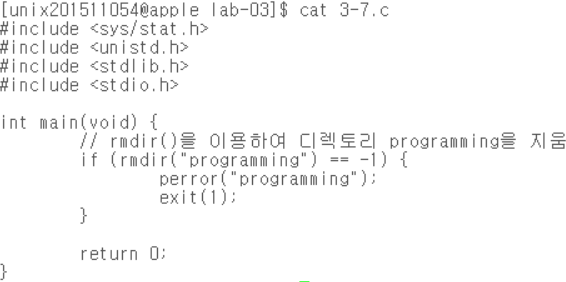
아래의 소스 코드처럼 rename 함수를 사용해서 linux 폴더를 LINUX 폴더로 수정하였습니다.



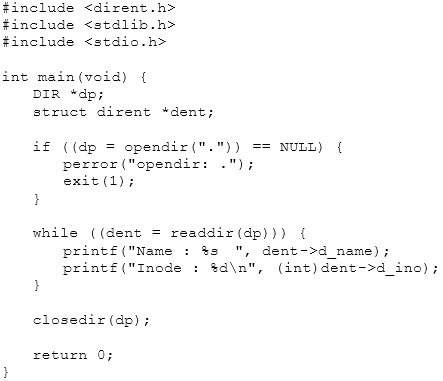
7. (rmdir() syscall, 파일 “3-7.c”) 시스템 콜 rmdir()을 이용하여 디렉토리 programming을 지우는 프로그램 3-7.c 를 작성하고 컴파일, 수행하시오. $ ls –l로 결과를 확인하시오.

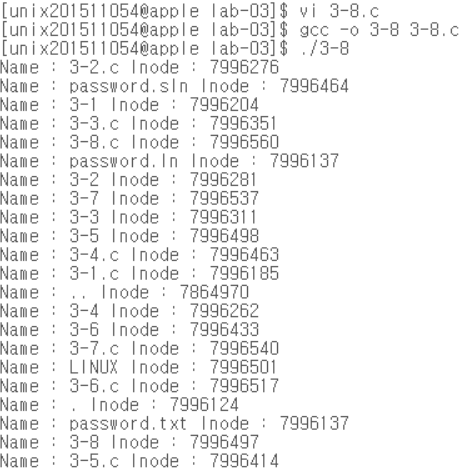


아래의 소스 코드처럼 rmdir 함수를 사용해서 programming 폴더를 삭제하였습니다.



8. (opendir(), readdir(), 파일 “3-8.c”) 다음과 같이 현재 디렉토리의 내용을 읽어 출력하는 프로그 램 3-8.c를 작성하고 컴파일하여 3-8을 만들고 수행하여 그 결과를 확인하시오.



파일 별 Inode가 적혀있습니다.

링크가 연결되어 있는 password.txt, password.ln은 inode가 같다는 것을 볼 수 있습니다.

끝.