

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1, 2 편

37 장. chmod 편

37 장. chmod 목 차 (책 실제페이지 : 200)

- 실무명령어# 37-1절 : 파일과 디렉토리의 퍼미션(permission) 정복하기
- 실무명령어# 37-2절 : 8진수로 파일의 퍼미션 설정하기
- 실무명령어# 37-3절 : 특정 문자로 퍼미션 설정하기
- 실무명령어# 37-4절 : 여러개의 파일과 디렉토리의 퍼미션 동시에 설정하기
- 실무명령어# 37-5절 : 모든파일과 디렉토리의 퍼미션 한번에 설정하기
- 실무명령어# 37-6절 : 특정디렉토리내의 모든파일과 서브디렉토리퍼미션 한번에 설정하기
- 실무명령어# 37-7절 : 특정파일에 SetUID Bit 설정하기
- 실무명령어# 37-8절 : 특정 파일에 SetGID bit 설정하기
- 실무명령어# 37-9절 : 특정 디렉토리에 Sticky Bit 설정하는 퍼미션 설정하기

- 본 자료의 저작권은 “슈퍼유저코리아” (www.superuser.co.kr)에 있습니다 .
- 본자료는 “슈퍼유저코리아”에서 출판한 『리눅스 실무명령어 바이블(1편,2편)』의 실제 내용중 일부분으로서 책의 본문내용을 확인하고자하시는 분들의 요청에 의해 요청횟수가 많았던 부분만을 일부 공개한 것입니다.
- 본 책자의 자세한 모든 정보는 <http://www.superuser.co.kr/linuxcommandbible/> 에서 확인하시기 바랍니다.
- 본 자료의 수정 및 편집은 허용되지않습니다.
- 본 책은 권수 : 2 권 (1 편, 2 편), 페이지수 : 총 1,859 페이지, 장(chapter): 382 장, 절: 1,900 개의 절로 구성되어있습니다.
- 저작권 및 문의전화(사무실) : TEL : 051-583-8128, FAX : 051-583-8130

실무명령어# 37-1 : 파일과 디렉토리의 퍼미션(permission) 정복하기

필자는 이번 장을 기회삼아 이 책을 구입하신 독자분들께 파일과 디렉토리에 대한 퍼미션(Permission)이라는 것에 대하여 정확하고 분명하게 설명하고자 작정을 하였습니다. 그만큼 중요하기 때문입니다. 즉, 이 부분에 대한 설명이 좀 길더라도 하나하나씩 차례대로 짚어가면서 실무적인 관점에서 이해하시기 바랍니다. 수많은 책과 강좌에서 파일의 퍼미션에 대한 설명을 이미 몇번을 접해 보셨으리라 생각합니다. 물론 완벽하게 이해한 분들도 많이 계시겠지만 아직 퍼미션에 대한 이해가 조금이라도 부족하신 분이라면 이번에 필자에게 기회를 주시고 이번 장의 설명내용을 정확하고 분명하게 이해하도록 하십시오. 필자 또한 이번 장에서 퍼미션에 대한 정확한 이해를 전달하기 위하여 무척 많은 노력을 하였습니다.

chmod 명령어는 특정 파일 또는 디렉토리의 퍼미션(permission)을 변경하는 명령어입니다. 가장 기본적인 리눅스명령어로서 리눅스 사용자라면 누구나 알고 있어야하는 명령어입니다. 퍼미션이란 특정 파일이나 디렉토리에 대하여 읽기, 기록하기, 삭제하기등의 권한을 설정해 놓은 것으로 다중사용자 운영체제(Multi-User Operating System)에서 파일의 접근권한과 보호등을 위하여 반드시 필요한 것입니다.

또한 퍼미션은 리눅스뿐아니라 UNIX, Solaris, AIX, FreeBSD 등의 운영체제에서도 같은 목적으로 사용할 수 있습니다. 특히, 웹(www)으로 홈페이지를 보여 줄 때에 웹문서파일의 퍼미션설정이 되어있지 않아서 "이 문서는 허용되지 않았습니다."라는 웹페이지 에러를 자주 접하곤 합니다. 이에대한 원인은 거의 대부분 웹페이지 파일의 퍼미션이 허용되어있지 않기



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

때문입니다.

홈페이지를 만들어서 리눅스나 유닉스서버등에 제작한 홈페이지파일을 서버로 업로드(Upload)한 후에 파일의 퍼미션설정하는 예를 많이 보셨으리라 생각합니다. 이는 다른 일반 사용자들에게 웹문서를 보여주기 위하여 퍼미션을 적절히 설정하는 것입니다.

필자가 설명하는 이 chmod 명령어의 설명을 잘 이해하시고 파일의 퍼미션을 목적에 맞게 설정하는 방법을 확실하게 익히시기 바랍니다.

파일(디렉토리)의 퍼미션(권한, permission) 종류에는 아래와 같이 5 가지가 있습니다. 아래 사항은 매우 중요하므로 꼭 기억해 두시기 바라며 특히 동일한 퍼미션이 파일에 적용될 때와 디렉토리에 적용될 때의 의미가 다르다는 것을 정확하게 이해하시기 바랍니다.

퍼미션	일반적의의미	파일	디렉토리
r	읽기(read)권한	파일을 읽을 수 있음	디렉토리의 내용을 볼수 있음 (ls 로 파일리스트 확인가능)
w	쓰기(write)권한	파일에 저장, 삭제할 수 있음	디렉토리에 파일저장, 디렉토리이름변경, 삭제등 가능
x	실행(execution)권한	파일을 실행할 수 있음	디렉토리에 접근할 수 있음 (cd 로 접근가능, ls 로 리스트는 불가)
s	SetUID, SetGID 권한	파일소유자(SetUID), 그룹소유자(SetGID)의 권한으로 실행함	
t	Sticky Bit 권한	공유디렉토리로 사용됨	

위의 예를 보시면 같은 권한일 경우에도 파일에 대한 의미와 디렉토리에 대한 의미에 있어서 다소 차이가 있음을 알 수 있습니다. 거듭 잔소리해서 죄송하지만 위의 의미를 정확하게 이해하지 못한다면 chmod 명령어의 정확한 사용과 실무활용이 어렵다는 점을 아셔야합니다.

명령어위치 : /bin/chmod

사용형식 : chmod [옵션] 8 진수퍼미션설정 대상파일들
 chmod [옵션] 퍼미션설정모드 대상파일들

가능한 옵션들

- c : 올바르게 변경된 파일들만 -v 옵션을 적용하여 자세히 보여준다.
- f : 가능한 불필요한 메시지를 보여주지 않고 간략하게 보여준다.
- v : 실행과정을 자세하게 보여준다.
- R : 디렉토리와 그안에 존재하는 서브디렉토리들까지 모두 적용한다.

파일의 퍼미션을 정확하게 공부하기 위해서는 2 진수, 8 진수, 10 진수의 관계를 정확하게 이해하셔야 합니다. 꼭 필요한 것은 아니지만 정확한 공부를 위하여 익혀두시기 바랍니다.

2 진수	8 진수	표시되는 퍼미션	의미하는 내용
------	------	----------	---------



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

000	0	---	아무런 권한이 없음
001	1	--x	실행(eXecution)권한만 있음
010	2	-w-	쓰기(Write)권한만 있음
011	3	-wx	쓰기와 실행권한만 있음
100	4	r--	읽기(Read)권한만 있음
101	5	r-x	읽기와 실행권한만 있음
110	6	rw-	읽기와 쓰기권한만 있음
111	7	rwX	읽기,쓰기,실행권한이 있음

이외에도 2 진수, 8 진수, 10 진수, 16 진수까지 정확하게 알고 있는 것이 좋습니다. 적어도 위의 2 진수와 8 진수의 관계에 대해서는 정확하게 알고 계셔야 chmod 에서 퍼미션 설정을 정확하게 할 수 있습니다.

보다 정확한 설명을 위하여 아래와 같이 "ls -l"의 결과를 보면서 계속 설명 드리겠습니다.

```
[root@file inc]# ls -l
total 52
-rw-r--r-x    1 working  working    1308 Nov  8 17:42 pop_board.php
-rwxr--r--    1 working  working    1550 Nov  8 17:42 pop_management.php
-rw-r--r--    1 working  working    1448 Nov  8 17:42 pop_menu1.php
-rwxr--r--    1 working  working    2634 Nov  8 17:42 pop_menu.php
-rw-r--r--    1 working  working   21340 Nov  8 17:42 public.js
-rw-r--r--    1 working  working    2499 Nov  8 17:42 quick1.php
-rw-rwxr--    1 working  working    2482 Nov  8 17:42 quick.php
-rw-r--r-x    1 working  working    2982 Nov  8 17:42 right_menu.php
[root@file inc]#
```

위의 예에서 각 행들의 첫번째 부분에 퍼미션(예: -rw-r--r-x)설정이 되어 있는 것을 보실 수 있습니다. 이 부분을 설정하기 위하여 chmod 라는 명령어를 사용합니다.

직접 설정하기에 앞서 이부분의 정확한 의미를 설명드리도록 하겠습니다. 위의 예에서 첫번째 pop_board.php 라는 파일의 정보는 다음과 같습니다.

```
-rw-r--r-x    1 working  working    1308 Nov  8 17:42 pop_board.php
```

- 파일퍼미션 : -rw-r--r-x
- 파일명 : pop_board.php
- 파일소유자 : working
- 파일그룹소유자 : working
- 파일사이즈 : 1308 byte
- 파일최종변경일시 : 11월 8일 17시 42분

위와 같은 정보에서 퍼미션부분을 보면 다음과 같습니다.

-	r	w	-	r	-	-	r	-	x
A. 파일유형	B. 소유자권한부분	C. 그룹소속자권한부분	D. 일반다른사용자권한부분						

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

A. 파일유형

이 위치에는 다음과 같은 문자들이 설정될 수 있으며 문자들의 의미는 다음과 같습니다.

- : 일반파일임을 의미함.
- b : 블럭구조의 특수파일(Block Special File)을 의미함.(예, /dev/sda)
- c : 입출력에 사용되는 특수파일(Character Special File)을 의미함.(예, /dev/console)
- d : 디렉토리(Directory)를 의미하며, 디렉토리도 하나의 특수파일로 취급됨.
- l : 심벌릭링크파일(Link File)임을 의미함.
- p : 파이프파일(Pipe)임을 의미함.
- s : 소켓파일(Socket)임을 의미함.

B. 소유자권한부분

이 위치에는 다음과 같이 8 가지의 경우로 설정될 수 있으며 그 각각의 의미는 다음과 같습니다.

- : 파일의 소유자에게 아무런 권한이 없음(8 진수표시:0, 2 진수표시:000)
- x : 파일의 소유자에게 실행(eXecution)권한만 있음(8 진수표시:1, 2 진수표시:001)
- w- : 파일의 소유자에게 쓰기(write)권한만 있음(8 진수표시:2, 2 진수표시:010)
- wx : 파일의 소유자에게 쓰기와 실행권한만 있음(8 진수표시:3, 2 진수표시:011)
- r-- : 파일의 소유자에게 읽기(Read)권한만 있음(8 진수표시:4, 2 진수표시:100)
- r-x : 파일의 소유자에게 읽기와 실행권한만 있음(8 진수표시:5, 2 진수표시:101)
- rw- : 파일의 소유자에게 읽기와 쓰기권한만 있음(8 진수표시:6, 2 진수표시:110)
- rwx : 파일의 소유자에게 읽기,쓰기,실행권한이 있음(8 진수표시:7, 2 진수표시:111)

위의 예에서는 이부분의 설정이 rw-로 되어 있으므로 working 이라는 파일소유자에게 읽기와 쓰기권한이 있음을 알수가 있습니다.

C. 그룹소속자권한부분

이 위치에는 다음과 같이 8 가지의 경우로 설정될 수 있으며 그 각각의 의미는 다음과 같습니다.

- : 파일의 그룹소속자에게 아무런 권한이 없음(8 진수표시:0, 2 진수표시:000)
- x : 파일의 그룹소속자에게 실행(eXecution)권한만 있음
(8 진수표시:1, 2 진수표시:001)
- w- : 파일의 그룹소속자에게 쓰기(write)권한만 있음(8 진수표시:2, 2 진수표시:010)
- wx : 파일의 그룹소속자에게 쓰기와 실행권한만 있음(8 진수표시:3, 2 진수표시:011)
- r-- : 파일의 그룹소속자에게 읽기(Read)권한만 있음(8 진수표시:4, 2 진수표시:100)
- r-x : 파일의 그룹소속자에게 읽기와 실행권한만 있음(8 진수표시:5, 2 진수표시:101)
- rw- : 파일의 그룹소속자에게 읽기와 쓰기권한만 있음(8 진수표시:6, 2 진수표시:110)
- rwx : 파일의 그룹소속자에게 읽기,쓰기,실행권한이 있음(8 진수표시:7, 2 진수표시:111)

위의 예에서 r--로 설정되어 있으므로 working 이라는 그룹에 소속된 사용자들에게 읽기권



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

한만 주어졌음을 알 수가 있습니다.

D. 일반다른사용자권한부분 (다음과 같은 8 가지로 설정될 수 있음)

- : 다른사용자들에게 아무런 권한이 없음(8 진수표시 : 0, 2 진수표시 : 000)
- x : 다른사용자들에게 실행(eXecution)권한만 있음(8 진수표시 : 1, 2 진수표시 : 001)
- w- : 다른사용자들에게 쓰기(write)권한만 있음(8 진수표시 : 2, 2 진수표시 : 010)
- wx : 다른사용자들에게 쓰기와 실행권한만 있음(8 진수표시 : 3, 2 진수표시 : 011)
- r-- : 다른사용자들에게 읽기(Read)권한만 있음(8 진수표시 : 4, 2 진수표시 : 100)
- r-x : 다른사용자들에게 읽기와 실행권한만 있음(8 진수표시 : 5, 2 진수표시 : 101)
- rw- : 다른사용자들에게 읽기와 쓰기권한만 있음(8 진수표시 : 6, 2 진수표시 : 110)
- rwx : 다른사용자들에게 읽기,쓰기,실행권한이 있음(8 진수표시 : 7, 2 진수표시 : 111)

위의 예에서 r-x 로 설정되어 있으므로 일반 다른 사용자들에게 읽기와 실행권한이 있음을 알 수가 있습니다.

위의 관계들만 정확하게 이해된다면 chmod 로 퍼미션을 설정하는 것은 다 배운것이나 다름 없습니다.

지금까지 파일의 퍼미션의 의미에 대해서 알아보았으므로 이제 파일의 퍼미션을 설정하는 방법에 대해서 설명하도록 하겠습니다.

즉, 파일의 퍼미션을 설정하는 방법에는 두가지방법이 있습니다.

첫번째 8 진수의 숫자로 지정하는 방법입니다.

예) chmod 755 a_file

==> 파일소유자에게 7 (2 진수로는 111)의 권한 즉, rwx 의 권한을 부여함.

==> 파일그룹소속자에게 5 (2 진수로는 101)의 권한 즉, r-x 의 권한을 부여함.

==> 다른사용자들에게 5 (2 진수로는 101)의 권한 즉, r-x 의 권한을 부여함.

이렇게 숫자로 지정을 하면 첫번째 숫자는 파일의 소유자권한, 두번째 숫자는 그룹소속자권한, 세번째 숫자는 다른사용자에 대한 권한을 각각 표시하게 됩니다.

두번째로는 특정문자를 이용한 퍼미션설정 방법입니다.

예) chmod o+rw a_file

==> a_file 에 대해 일반다른사용자들(others)에게 읽기(r, Read)권한과 쓰기(w, Write)권한을 부여함.

위와 같은 의미가 됩니다. 즉, 특정문자로 사용하는 종류에는 다음과 같은 것들이 있습니다.

- . u : 소유자(user) 를 의미함
- . g : 그룹(group) 을 의미함
- . o : 다른사용자(others) 들을 의미함
- . a : 모두(all) 를 의미함(Default)



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

그리고 이 문자와 함께 "+"기호가 사용되면 "권한을 부여한다"라는 의미이며, "-"기호가 사용되면 "권한을 제거한다"라는 의미가 됩니다. 즉, 위의 예(chmod o+rw a_file)는 이와 같이 해석이 된 것입니다.

이에 대한 예는 아래 사용예들을 보시면서 정확한 이해를 하시기 바랍니다.

실무명령어# 37-2 : 8 진수로 파일의 퍼미션 설정하기

testfile 의 현재 퍼미션은 다음과 같습니다. 이 퍼미션을 다양하게 각각 설정해 보도록 하겠습니다.

```
[root@file test]# ls -l testfile
-----x   1 root   root           0 Nov  2 04:42 testfile
[root@file test]#
```

아래는 퍼미션을 755 로 설정 하였습니다. 즉, 파일의 소유자에게는 모든 권한(읽기,쓰기,실행)을 부여(7)하고, 그룹소속자에게는 읽기와 실행권한을 부여(5)하였으며, 다른사용자에게는 읽기와 실행권한을 부여(5)하였습니다.

이 설정은 웹문서파일들에 설정하는 일반적인 퍼미션입니다.

```
[root@file test]# chmod 755 testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwxr-xr-x   1 root   root           0 Nov  2 04:42 testfile
[root@file test]#
```

그리고 다음은 testfile 의 퍼미션을 644 로 설정 하였습니다. 즉, 파일의 소유자에게는 읽기와 쓰기권한을 부여(6)하고 그룹소속자에게는 읽기권한을 부여(4)하고 다른사용자들에게는 읽기권한만을 부여(4)하였습니다.

이 퍼미션 설정(644) 또한 웹문서파일의 일반적인 권한설정으로 주로 사용됩니다.

```
[root@file test]# chmod 644 testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rw-r--r--   1 root   root           0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 600 으로 testfile 의 퍼미션을 설정 하였습니다. 즉, testfile 의 소유자에게는 읽기와 쓰기의 권한을 부여(6)하고 그룹소속자와 다른 사용자에게는 아무런 권한도 부여(0)하지 않았습니다.

이런 권한설정은 보안이 요구되는 파일에 사용되는 퍼미션설정입니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

```
[root@file test]# chmod 600 testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rw-----  1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 testfile 의 퍼미션을 700 으로 설정 하였습니다. 즉, 파일의 소유자에게는 모든 권한(읽기,쓰기,실행)을 부여(7)하고 그룹소속자와 다른 사용자에게는 아무런 권한도 부여(0)하지 않았습니다.

이런 권한설정 또한 보안이 요구되는 파일에 사용되는 퍼미션 설정입니다.

```
[root@file test]# chmod 700 testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwx-----  1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

실무명령어# 37-3 : 특정 문자로 퍼미션 설정하기

testfile 의 퍼미션이 아래와 같이 설정되어 있습니다.

```
[root@file test]# ls -l
total 0
-rwx-----  1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

아래의 예는 testfile 에 대하여 모두(all 즉, 소유자,그룹소속자,다른사용자들)에게 읽기(r)권한을 부여(+)하였습니다. 결과 변경된 퍼미션을 확인해 보시면 소유자,그룹소속자,다른사용자들에게 모두 읽기(Read)권한이 부여되었음을 확인하실 수 있습니다.

```
[root@file test]# chmod a+r testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwxr--r--  1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 testfile 에 대하여 모두(all, 즉, 소유자,그룹소속자,다른사용자들)에게 쓰기(w) 권한을 부여(+)하였습니다. 결과 변경된 퍼미션을 확인해 보시면 소유자,그룹소속자,다른사용자들에게 모두 쓰기(Write)권한이 부여되었음을 확인하실 수 있을 것입니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

```
[root@file test]# chmod a+w testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwxrwx-rwx    1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 testfile 에 대하여 모두(all, 즉, 소유자,그룹소속자,다른사용자들)에게 실행(eXecution)권한을 부여(+)하였습니다. 결과 변경된 퍼미션을 확인해 보시면 소유자,그룹소속자,다른사용자들에게 모두 실행(eXecution)권한이 부여되었음을 확인하실 수 있을 것입니다.

```
[root@file test]# chmod a+x testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwxrwxrwx    1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 testfile 에 대하여 모두(all, 즉, 소유자,그룹소속자,다른사용자들)에게 읽기(r), 쓰기(w), 실행(x)권한을 제거(-)한 예입니다. 결과 변경된 퍼미션을 확인해 보시면 소유자, 그룹소속자, 다른사용자들에게 어떠한 권한도 주어지지 않았음을 확인하실 수 있을 것입니다.

```
[root@file test]# chmod a-rwx testfile
[root@file test]#
[root@file test]# ls -l
total 0
-----    1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 testfile 에 대하여 소유자(user)에게만 읽기(r), 쓰기(w), 실행(x)권한을 부여(+)한 예입니다. 결과 변경된 퍼미션을 확인해 보시면 소유자에게 읽기(r), 쓰기(w), 실행(x)권한이 부여된 것을 확인할 수 있습니다.

```
[root@file test]# chmod u+rwx testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwx-----    1 root    root          0 Nov  2 04:42 testfile
[root@file test]#
```

다음에는 testfile 에 대하여 그룹소속자(group)에게 읽기(r),와 실행(x)권한을 부여(+)한 예입니다. 결과 변경된 퍼미션을 확인해 보시면 그룹소속자에게 읽기(r),와 실행(x)권한이 부여된 것을 확인할 수 있습니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

```
[root@file test]# chmod g+rx testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwxr-x--- 1 root root 0 Nov 2 04:42 testfile
[root@file test]#
```

다음에는 testfile 에 대하여 다른사용자(others)들에게 실행(x)권한을 부여(+)한 예입니다. 결과 변경된 퍼미션을 확인해 보시면 다른사용자에게 실행(x)권한이 부여된 것을 확인 하실 수 있습니다.

```
[root@file test]# chmod o+x testfile
[root@file test]#
[root@file test]# ls -l
total 0
-rwxr-x--x 1 root root 0 Nov 2 04:42 testfile
[root@file test]#
```

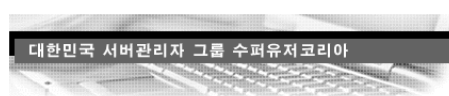
실무명령어# 37-4 : 여러개의 파일과 디렉토리의 퍼미션 동시에 설정하기

지금까지의 예들은 모두 하나의 파일을 대상으로 퍼미션을 설정하였습니다. 이번에는 동시에 여러개의 파일퍼미션을 설정하는 예를 보도록 하겠습니다.

아래의 예를 보시면 현재 위치한 디렉토리에 여러개의 파일 또는 디렉토리가 존재합니다.

```
[root@file Applications]# ls -l
total 28504
-rw-r--r-- 1 root root 525124 May 1 2003 PERL_DBI.tar.gz
-rw-r--r-- 1 root root 3413829 Oct 19 2002 php-4.2.3.tar.gz
-rw-r--r-- 1 root root 4501159 Apr 19 2003 php-4.3.0.tar.gz
-rw-r--r-- 1 root root 4521992 Jun 20 08:07 php-4.3.2.tar.gz
-rw-r--r-- 1 root root 4565509 Sep 5 08:38 php-4.3.3.tar.gz
drwxr-xr-x 10 root root 4096 Nov 7 21:21 proftpd-1.2.8
-rw-r--r-- 1 root root 966281 Jun 2 10:33 proftpd-1.2.8.tar.gz
-rw-r--r-- 1 root root 2261992 Oct 12 2002 qpopper4.0.4.tar.gz
-rw-r--r-- 1 root root 1374835 Jun 8 2002 sendmail.8.11.6.tar.gz
-rw-r--r-- 1 root root 1834935 Jun 8 2002 sendmail.8.12.2.tar.gz
-rw-r--r-- 1 root root 46641 Jul 16 05:08 sendmail.cf
-rw-r--r-- 1 root root 18244 Apr 4 2003 services
[root@file Applications]#
```

여기서 php 로 시작하는 모든 파일의 퍼미션을 755 로 수정하고자 합니다. 따라서 "chmod 755 php*"라는 명령으로서 php 로 시작하는 모든 파일과 디렉토리의 퍼미션을 755 로 수정하였습니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

```
[root@file Applications]# chmod 755 php*
[root@file Applications]#
[root@file Applications]# ls -l
total 28504
-rw-r--r-- 1 root root 525124 May 1 2003 PERL_DBI.tar.gz
-rwxr-xr-x 1 root root 3413829 Oct 19 2002 php-4.2.3.tar.gz
-rwxr-xr-x 1 root root 4501159 Apr 19 2003 php-4.3.0.tar.gz
-rwxr-xr-x 1 root root 4521992 Jun 20 08:07 php-4.3.2.tar.gz
-rwxr-xr-x 1 root root 4565509 Sep 5 08:38 php-4.3.3.tar.gz
drwxr-xr-x 10 root root 4096 Nov 7 21:21 proftpd-1.2.8
-rw-r--r-- 1 root root 966281 Jun 2 10:33 proftpd-1.2.8.tar.gz
-rw-r--r-- 1 root root 2261992 Oct 12 2002 qpopper4.0.4.tar.gz
-rw-r--r-- 1 root root 1374835 Jun 8 2002 sendmail.8.11.6.tar.gz
-rw-r--r-- 1 root root 1834935 Jun 8 2002 sendmail.8.12.2.tar.gz
-rw-r--r-- 1 root root 46641 Jul 16 05:08 sendmail.cf
-rw-r--r-- 1 root root 18244 Apr 4 2003 services
[root@file Applications]#
```

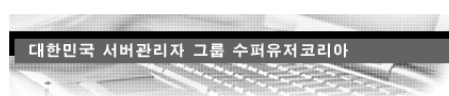
이렇게 chmod 에는 여러개의 파일을 동시에 동일한 퍼미션으로 설정할 수 있습니다. 서버 관리를 하다보면 특정파일 하나에 대해서 퍼미션을 설정하는 경우도 많이 발생하지만 이번 예와같이 여러개의 파일을 동시에 변경하는 경우도 종종 있습니다.

실무명령어# 37-5 : 모든파일과 디렉토리의 퍼미션 한번에 설정하기

다음은 현재 디렉토리에 존재하는 모든 파일과 디렉토리의 퍼미션을 동시에 설정하는 예입니다. 현재 디렉토리에는 보시는 바와 같이 여러개의 파일이 존재합니다.

```
[root@file admin]# ls -l
total 48
-rw-r--r-- 1 working working 2697 Nov 8 17:42 admin.php
-rw-r--r-- 1 working working 2345 Nov 8 17:42 counter_admin.php
-rw-r--r-- 1 working working 507 Nov 8 17:42 del_ok.php
-rw-r--r-- 1 working working 1775 Nov 8 17:42 del.php
-rw-r--r-- 1 working working 1206 Nov 8 17:42 insert.php
-rw-r--r-- 1 working working 2368 Nov 8 17:42 kt_poll.html
-rw-r--r-- 1 working working 4315 Nov 8 17:42 kt_poll_list.php
-rw-r--r-- 1 working working 614 Nov 8 17:42 update_ok.php
-rw-r--r-- 1 working working 2903 Nov 8 17:42 update.php
-rw-r--r-- 1 working working 1588 Nov 8 17:42 view.php
-rw-r--r-- 1 working working 1466 Nov 8 17:42 vote_insert.php
[root@file admin]#
```

현재 디렉토리에 존재하는 모든 파일들의 퍼미션을 동시에 모두 700 으로 설정 하였습니다. 그런 다음 "ls -l"로 변경된 퍼미션설정을 확인한 것입니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

```
[root@file admin]# chmod 700 *
[root@file admin]#
[root@file admin]# ls -l
total 48
-rwx----- 1 working working 2697 Nov  8 17:42 admin.php
-rwx----- 1 working working 2345 Nov  8 17:42 counter_admin.php
-rwx----- 1 working working  507 Nov  8 17:42 del_ok.php
-rwx----- 1 working working 1775 Nov  8 17:42 del.php
-rwx----- 1 working working 1206 Nov  8 17:42 insert.php
-rwx----- 1 working working 2368 Nov  8 17:42 kt_poll.html
-rwx----- 1 working working 4315 Nov  8 17:42 kt_poll_list.php
-rwx----- 1 working working  614 Nov  8 17:42 update_ok.php
-rwx----- 1 working working 2903 Nov  8 17:42 update.php
-rwx----- 1 working working 1588 Nov  8 17:42 view.php
-rwx----- 1 working working 1466 Nov  8 17:42 vote_insert.php
[root@file admin]#
```

실무명령어# 37-6 : 특정디렉토리내의 모든파일과 서브디렉토리퍼미션 한번에 설정하기

다음은 특정 디렉토리내에 존재하는 모든 파일과 서브디렉토리내의 모든 파일과 디렉토리까지 한꺼번에 퍼미션을 설정하는 예를 보겠습니다.

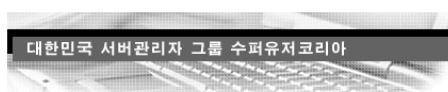
현재 디렉토리내에는 보다시피 www 라는 디렉토리가 존재합니다. 지금 www 라는 디렉토리와 그 안에 존재하는 모든 파일과 디렉토리의 퍼미션을 한번에 755 로 동일하게 설정 하였습니다.

이 경우 서브디렉토리내의 모든 파일과 디렉토리까지도 한번에 동일한 퍼미션을 설정하고자 한다면 아래의 예와 같이 -R 옵션을 사용하셔야 합니다.

```
[root@file bible]# ls -l
total 8
d--x----- 2 bible bible 4096 Nov 18 16:06 test
drwxr-xr-x  2 bible bible 4096 Nov  8 19:43 www
[root@file bible]#
[root@file bible]# chmod -R 755 www
[root@file bible]#
[root@file bible]# ls -l
total 8
d--x----- 2 bible bible 4096 Nov 18 16:06 test
drwxr-xr-x  2 bible bible 4096 Nov  8 19:43 www
[root@file bible]#
```

위의 결과로 www 디렉토리자체의 퍼미션과 www 내에 존재하는 파일들과 서브디렉토리들까지 모두 755 로 퍼미션이 변경 되었습니다.

실무명령어# 37-7 : 특정파일에 SetUID Bit 설정하기



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

이번에는 특정파일에 SetUID Bit 를 설정하는 방법에 대해서 설명 하겠습니다. 특정 파일에 SetUID Bit 가 설정되어 있다면 다른 사용자들이 그 파일을 실행하였을 경우 실행되는 동안에는 실행시킨 사용자의 권한이 아닌 파일의 소유자권한으로 실행이 됩니다.

이 퍼미션을 설정하는 방법은 4755, 4644, 4750 등과 같이 지금까지 설명한 일반퍼미션앞에 4 를 붙이시면 됩니다.

아래의 예는 testfile 에 일반 퍼미션 755 와 함께 SetUID Bit(4)를 설정한 예입니다.

```
[root@file bible]# ls -l
total 4
-rw-r--r--  1 root    root      71 Nov 18 18:17 testfile
[root@file bible]#
[root@file bible]# chmod 4755 testfile
[root@file bible]#
[root@file bible]# ls -l
total 4
-rwsr-xr-x  1 root    root      71 Nov 18 18:17 testfile
[root@file bible]#
```

위의 결과를 보시면 testfile 의 퍼미션설정의 소유자 실행퍼미션 자리에 s 라는 문자가 설정된 것을 보실 수 있을 것입니다. 파일의 소유자 실행퍼미션 자리에 s 라는 문자가 있다면 다른 사용자들이 이 파일을 실행하더라도 파일의 소유자권한으로 실행이 됩니다.

참고로 다음의 예에서 설명하는 SetGID 는 일반퍼미션앞에 2 를 붙이시면 되고 Sticky Bit 는 1 을 붙이시면 됩니다.

실무명령어# 37-8 : 특정 파일에 SetGID bit 설정하기

이번에는 SetGID 를 설정해 보도록 하겠습니다. 앞의 SetUID 와 동일한 방법으로 설정하지만 일반 퍼미션앞에 4 가 아니라 2 를 붙이면 SetGID 가 설정됩니다.

아래의 예는 testfile 에 일반퍼미션 755 와 함께 SetGID Bit(2)를 설정한 예입니다.

```
[root@file bible]# ls -l
total 4
-rw-r--r--  1 root    root      71 Nov 18 18:17 testfile
[root@file bible]#
[root@file bible]# chmod 2755 testfile
[root@file bible]#
[root@file bible]# ls -l
total 4
-rwxr-sr-x  1 root    root      71 Nov 18 18:17 testfile
[root@file bible]#
```

chmod 명령어가 실행된 후에 testfile 의 그룹소유자의 실행퍼미션 자리를 보시면 s 라는 문자가 설정된 것을 확인할 수 있습니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1,11 편

그룹소유자의 실행퍼미션자리에 s로 설정되면 SetGID로 설정되었음을 의미합니다. SetGID가 설정된 파일을 다른 사용자가 실행을 시키면 실행시킨 사용자의 권한으로 파일실행이 되는 것이 아니라 파일의 소유자그룹권한으로 실행이 됩니다.

실무명령어# 37-9 : 특정 디렉토리에 Sticky Bit 설정하는 퍼미션 설정하기

이번에는 특정 디렉토리에 Sticky Bit 를 설정하는 예를 보도록 하겠습니다. Sticky Bit 라고 하면 조금은 생소하게 느끼시는 독자분이 계실지는 모르겠습니다.

리눅스에서는 흔히 /tmp 디렉토리를 Sticky Bit 로 설정 합니다. 이 /tmp 디렉토리내에서 특정 파일을 생성하면 생성한 사용자의 소유로 파일이 생성됩니다.

그리고 다른 사용자가 /tmp 디렉토리에 다른 파일을 생성하면 역시 생성한 사용자의 소유로 파일이 생성됩니다. 파일을 삭제할 때에도 생성한 사람만이 삭제를 할 수가 있습니다.

이런 디렉토리를 흔히 공유디렉토리라고 합니다. 즉, 파일생성을 아무나 할 수 있으며 생성된 파일은 각각 생성한 사람의 소유로 되며 수정, 삭제 또한 생성한 사람만이 그 파일을 수정하고, 지울 수 있도록 되어 있습니다. 이런 "공동장소"로 사용되는 디렉토리설정을 하기 위하여 Sticky Bit 를 설정합니다.

설정하는 방법은 일반퍼미션앞에 1을 붙이시면 됩니다.

다음의 예를 보시기 바랍니다.

```
[root@file bible]# ls -l
total 4
drwxr-xr-x  2 root    root      4096 Nov 18 18:29 testdir
[root@file bible]#
[root@file bible]# chmod 1755 testdir
[root@file bible]#
[root@file bible]# ls -l
total 4
drwxr-xr-t  2 root    root      4096 Nov 18 18:29 testdir
[root@file bible]#
```

위의 예에서 testdir 이라는 디렉토리의 퍼미션을 1755 로 설정하면서 Sticky Bit 를 부여하였습니다. 부여한 결과 다른사용자 실행 퍼미션자리에 보시면 t 라는 문자로 셋팅된 것을 확인하실 수 있습니다. 이 t 라는 문자의 의미가 Sticky Bit 라는 것을 의미합니다.

이제부터 testdir 이라는 디렉토리는 공유목적으로 사용할 수 있도록 Sticky Bit 설정이 된 것입니다.

즉, bible 이라는 사용자가 testdir 에 biblefile 이라는 파일을 생성하였다고 하면 이 biblefile 은 bible 이라는 사용자의 소유로 생성되고, 삭제와 변경도 bible 이라는 사용자만이 할 수가 있습니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부이지만 유용하게 사용하시기 바랍니다.

리눅스 실무명령어 바이블 1, 2 편

그리고 `sspark` 이라는 사용자가 `testdir` 에 `ssparkfile` 이라는 파일을 생성하였다고 하면 이 `ssparkfile` 은 `sspark` 이라는 사용자의 소유로 되고, 삭제와 변경도 `sspark` 이라는 사용자가 할 수가 있습니다. 물론 다른 사용자들도 모두 동일하게 적용이 됩니다.

하지만, Sticky Bit 가 설정된 디렉토리내의 파일들에 대하여 root 는 언제나 마음대로 변경, 삭제등이 가능합니다.

여러분 이제 파일과 디렉토리의 퍼미션에 대해서 어느정도 자신감이 생겼으리라 생각합니다. 파일과 디렉토리의 퍼미션설정에 대한 필자의 소임을 마치고 이제 실무에서 적용하는 것에 대해서는 여러분들께 맡기도록 하겠습니다. 부디 필자의 소망처럼 정확한 이해와 효율적인 사용으로 힘있고 강력한 서버관리자가 되시길 바랍니다.

- 본 자료의 저작권은 “슈퍼유저코리아” (www.superuser.co.kr)에 있습니다 .
- 본자료는 “슈퍼유저코리아”에서 출판한 『리눅스 실무명령어 바이블(1 편, 2 편)』의 실제 내용중 일부분으로서 책의 본문내용을 확인하고자하시는 분들의 요청에 의해 요청횟수가 많았던 부분만을 일부 공개한 것입니다.
- 본 책자의 자세한 모든 정보는 <http://www.superuser.co.kr/linuxcommandbible/> 에서 확인하시기 바랍니다.
- 본 자료의 수정 및 편집은 허용되지않습니다.
- 본 책은 권수 : 2 권 (1 편, 2 편), 페이지수 : 총 1,859 페이지, 장(chapter): 382 장, 절: 1,900 개의 절로 구성되어있습니다.
- 저작권 및 문의전화(사무실) : TEL : 051-583-8128, FAX : 051-583-8130

