chmod 빠른 참조

파일 퍼미션, 숫자/심볼 모드, 특수 비트, umask

숫자 모드

8진수 퍼미션 숫자

4	읽기 (r)
2	쓰기 (w)
1	실행 (x)
0	권한 없음

세 자리 형식

```
chmod 755 file    # rwxr-xr-x
chmod 644 file    # rw-r--r--
chmod 700 file    # rwx------
chmod 600 file    # rw-------
```

숫자 계산

7 (4+2+1)	rwx — 읽기, 쓰기, 실행
6 (4+2)	rw- — 읽기, 쓰기
5 (4+1)	r-x — 읽기, 실행
4	r-- — 읽기 전용
3 (2+1)	-wx — 쓰기, 실행
2	-w- — 쓰기 전용
1	--x — 실행 전용
0	--- — 권한 없음

심볼 모드

문법: [ugoa][+-=][rwxXst]

u	소유자 (user)
g	그룹
o	기타
a	전체 (u + g + o)
+	퍼미션 추가
-	퍼미션 제거
=	퍼미션 정확히 설정

심볼 예시

```
chmod u+x file         # owner: add execute
chmod g-w file         # group: remove write
chmod o=r file         # others: set read only
chmod a+r file         # all: add read
chmod u+x,g-w,o= file # combined operations
```

일반 퍼미션

파일 퍼미션 프리셋

644	**rw-r--r--**	기본 파일 — 소유자 rw, 기타 읽기
755	**rwxr-xr-x**	스크립트/바이너리 — 소유자 rwx, 기타 rx
600	**rw-------**	비공개 파일 — 소유자만
400	**r--------**	읽기 전용 비공개 (SSH 키)
666	**rw-rw-rw-**	누구나 쓸 수 있는 파일 (비권장)
777	**rwxrwxrwx**	모두 전체 권한 (비권장)

빠른 참조

```
chmod 644 *.html       # web files: owner rw, world r
chmod 755 *.sh         # scripts: owner rwx, world rx
chmod 600 ~/.ssh/id_*  # SSH keys: owner only
chmod 400 secret.pem   # certificate: read-only
```

디렉토리 퍼미션

디렉토리에서 퍼미션의 의미

r (4)	디렉토리 내용 목록 (**ls**)
w (2)	디렉토리에서 파일 생성/삭제
x (1)	디렉토리 접근 (cd)
rx (5)	목록 + 접근 (일반적인 읽기)
rwx (7)	전체 권한

일반 디렉토리 퍼미션

```
chmod 755 dir/     # standard: owner rwx, others rx
chmod 700 dir/     # private: owner only
chmod 750 dir/     # group access: owner rwx, group rx
chmod 1777 /tmp    # sticky bit: only owner can delete
```

특수 비트

Setuid, Setgid, Sticky

Setuid (4xxx)	파일 소유자로 실행 (예: **passwd**)
Setgid (2xxx)	파일 그룹으로 실행 / 디렉토리 그룹 상속
Sticky (1xxx)	소유자만 파일 삭제 가능 (예: **/tmp**)

특수 비트 설정

```
chmod 4755 program     # setuid: -rwsr-xr-x
chmod 2755 dir/        # setgid: drwxr-sr-x
chmod 1755 dir/        # sticky: drwxr-xr-t
chmod u+s program      # symbolic setuid
chmod g+s dir/         # symbolic setgid
chmod +t dir/          # symbolic sticky bit
```

재귀

재귀적 퍼미션 변경

```
chmod -R 755 dir/          # set all to 755 recursively
chmod -R u+rwX dir/        # owner rw, +x on dirs only
chmod -R go-w dir/         # remove group/other write
```

find로 파일/디렉토리 구분

```
# set directories to 755, files to 644
find /path -type d -exec chmod 755 {} +
find /path -type f -exec chmod 644 {} +
```

대문자 X — 조건부 실행

x (소문자)	모든 파일과 디렉토리에 실행 추가
X (대문자)	디렉토리와 이미 실행 가능한 파일에만 실행 추가

umask

umask 동작 방식

umask	현재 umask 표시
umask 022	파일: 644, 디렉토리: 755
umask 077	파일: 600, 디렉토리: 700
umask 002	파일: 664, 디렉토리: 775

umask 계산

```
# default permission minus umask = effective
# Files:  666 - 022 = 644 (rw-r--r--)
# Dirs:   777 - 022 = 755 (rwxr-xr-x)
umask          # display current umask
umask 022      # typical default
umask -S       # show in symbolic notation
```

일반 패턴

일상적인 사용 사례

Web root	**chmod -R 755 /var/www/html**
Config file	**chmod 600 app.conf**
SSH directory	**chmod 700 ~/.ssh**
SSH authorized_keys	**chmod 600 ~/.ssh/authorized_keys**
Shared directory	**chmod 2775 /shared** (setgid)
Log files	**chmod 640 /var/log/app.log**
Cron scripts	**chmod 755 /etc/cron.daily/myjob**
Temp directory	**chmod 1777 /tmp** (sticky)

퍼미션 확인

```
ls -l file.txt       # show permissions
ls -ld dir/          # show directory permissions
stat -c '%A %a %n' * # symbolic + numeric + name
getfacl file.txt     # show ACLs (if in use)
```