

CHMOD 快速参考

文件权限、数字与符号模式、特殊位、umask

数字模式

八进制权限位

4 读 (r)
2 写 (w)
1 执行 (x)
0 无权限

三位数格式

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

权限值计算

7 (4+2+1) rwx — 读、写、执行
6 (4+2) rw — 读、写
5 (4+1) r-x — 读、执行
4 r — 只读
3 (2+1) -wx — 写、执行
2 -w — 只写
1 -x — 只执行
0 --- 无权限

符号模式

语法: [ugoa][+=[rwxXst]

u 所有者 (user)
g 组 (group)
o 其他 (others)
a 所有 (u+g+o)
+ 添加权限
- 移除权限
= 精确设置权限

符号模式示例

```
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
```

常用权限

文件权限预设

644 **rw-r--r--** 默认文件——所有者读写, 其他只读
755 **rwxr-xr-x** 脚本/二进制——所有者全权, 其他读执行
600 **rw-----** 私密文件——仅所有者
400 **r-----** 只读私密文件 (SSH 密钥)
666 **rw-rw-rw-** 所有人可写文件 (避免使用)
777 **rwxrwxrwx** 所有人全权访问 (避免使用)

快速参考

```
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
```

目录权限

目录权限含义

r (4) 列出目录内容 (ls)
w (2) 在目录中创建 / 删除文件
x (1) 进入目录 (cd)
rx (5) 列出 + 进入 (典型只读)
rwx (7) 完全控制

常用目录权限

```
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
```

特殊位

Setuid, Setgid, Sticky

Setuid (4xxx) 以文件所有者身份运行 (如 passwd)
Setgid (2xxx) 以文件组运行 / 继承目录组
Sticky (1xxx) 仅所有者可删除文件 (如 /tmp)

设置特殊位

```
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
```

递归

递归修改权限

```
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
```

用 find 区分文件与目录

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

大写 X——条件执行位

x (小写) 对所有文件和目录添加执行位
X (大写) 仅对目录和已有执行位的文件添加执行位

umask

umask 工作原理

umask 显示当前 umask
umask 022 文件: 644, 目录: 755
umask 077 文件: 600, 目录: 700
umask 002 文件: 664, 目录: 775

umask 计算

```
# 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
```

常用场景

日常用例

```
Web 根目录 `chmod -R 755 /var/www/html`
配置文件 `chmod 600 app.conf`
SSH 目录 `chmod 700 ~/.ssh`
SSH authorized_keys `chmod 600 ~/.ssh/authorized_keys`
共享目录 `chmod 2775 /shared` (setgid)
日志文件 `chmod 640 /var/log/app.log`
Cron 脚本 `chmod 755 /etc/cron.daily/myjob`
临时目录 `chmod 1777 /tmp` (sticky)
```

查看权限

```
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)
```