

CHMOD クイックリファレンス

ファイルパーミッション、数値モード・記号モード、特殊ビット、umask

数値モード

8 進数パーミッションの桁

- 4 読み取り (r)
- 2 書き込み (w)
- 1 実行 (x)

0 パーミッションなし

3 桁の形式

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

桁の計算

- 7 (4+2+1) rwx 読み取り、書き込み、実行
- 6 (4+2) rw 読み取り、書き込み
- 5 (4+1) r-x 読み取り、実行
- 4 r-- 読み取りのみ
- 3 (2+1) -wx 書き込み、実行
- 2 -w- 書き込みのみ
- 1 --x 実行のみ
- 0 --- パーミッションなし

記号モード

構文: [ugoa][+|=][rwxXst]

- u ユーザー (オーナー)
- g グループ
- o その他
- a 全員 (u+g+o)
- + パーミッションを追加
- パーミッションを削除
- = パーミッションを設定

記号モードの例

```
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
```

よく使うパーミッション

ファイルパーミッションのプリセット

- 644 **rw-r--r--** 標準ファイル・オーナー rw、その他 r
- 755 **rwxr-xr-x** スクリプト/バイナリ・オーナー rwx、その他 rx
- 600 **rw-----** プライベートファイル・オーナーのみ
- 400 **r-----** 読み取り専用プライベート (SSH 鍵)
- 666 **rw-rw-rw-** 誰でも書き込み可 (避けること)
- 777 **rwxrwxrwx** 全員にフルアクセス (避けること)

クイックリファレンス

```
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
```

ディレクトリのパーミッション

ディレクトリにおけるパーミッションの意味

- r (4) ディレクトリの内容一覧 (ls)
- w (2) ディレクトリ内のファイル作成 / 削除
- x (1) ディレクトリへのアクセス (cd)
- rx (5) 一覧表示 + アクセス (読み取りの典型)
- rwx (7) フルコントロール

よく使うディレクトリパーミッション

```
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
```

特殊ビット

Setuid、Setgid、スティッキービット

- Setuid (4xxx) ファイルオーナーとして実行 (例: passwd)
- Setgid (2xxx) ファイルグループとして実行 / ディレクトリのグループ継承
- スティッキー (1xxx) オーナーのみがファイルを削除可能 (例: /tmp)

特殊ビットの設定

```
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
```

再帰的変更

再帰的なパーミッション変更

```
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
```

findを使ったファイルとディレクトリ分離

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

大文字 X - 条件付き実行ビット

- x (小文字) 全ファイルとディレクトリに実行を追加
- X (大文字) ディレクトリおよび既に実行可能なファイルにのみ追加

umask

umask の仕組み

umask 現在の umask を表示

umask 022	ファイル: 644、ディレクトリ: 755
umask 077	ファイル: 600、ディレクトリ: 700
umask 002	ファイル: 664、ディレクトリ: 775

umask の計算

```
# 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 -5 # show in symbolic notation
```

よく使うパターン

日常的なコースケース

Web ルート	chmod -R 755 /var/www/html`
設定ファイル	chmod 600 app.conf`
SSH ディレクトリ	chmod 700 ~/.ssh`
SSH authorized_keys	chmod 600 ~/.ssh/authorized_keys`
共有ディレクトリ	chmod 2775 /shared` (setgid)
ログファイル	chmod 640 /var/log/app.log`
cron スクリプト	chmod 755 /etc/cron.daily/myjob`
一時ディレクトリ	chmod 1777 /tmp` (スティッキー)

パーミッションの確認

```
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)
```