Unix File Permissions

Enjoy this cheat sheet at its fullest within Dash, the macOS documentation browser.

Permissions

Octal	Symbol	Permission
`0`	`---`	No permissions
`1`	`--x`	Execute
`2`	`-w-`	Write
`3`	`-wx`	Write and execute
`4`	`r--`	Read
`5`	`r-x`	Read and execute
`6`	`rw-`	Read and write
`7`	`rwx`	Read, write and execute

File Types

`-`	Regular file Example: `-rw-r--r-- 1 root 0 1 January 00:00 file`
`d`	Directory Example: `drwxr-xr-x 3 root staff 102 1 January 00:00`
`l`	Symbolic link Example: `lrwxrwxrwx 1 root root 4 1 January 00:00 rtc -> rtc0`

`b`	Block special device Example: `brw-rw---- 1 root disk 1 0 1 January 00:00 ram0`
`c`	Character device Example: `crw-rw-rw- 1 root root 1 3 1 January 00:00 null`
`s`	Unix socket Example: `srw-rw-rw- 1 root root 0 1 January 00:00 acpid.socket`
`p`	Named pipe Example: `prw-r--r-- 1 root root 0 1 January 00:00 pipe`

Special Mode Bits

`setuid` (Set User ID)

When the setuid permission is set on an executable file, a process that runs this file is granted access based on the owner of the file (usually root)

This special permission allows a user to access files and directories that are normally only available to the owner.

Example: The setuid permission on the passwd command makes it possible for a user to change passwords, assuming the permissions of the root ID:

```
-r-sr-sr-x 3 root sys 104580 Sep 16 12:02 /usr/bin/passwd
```

`setgid` (Set Group ID)

The set-group identification (setgid) permission is similar to setuid, except that the process's effective group ID (GID) is changed to the group owner of the file.

Example: The /usr/bin/mail command has setgid permissions:

```
-r-x--s--x 1 root mail 63628 Sep 16 12:01 /usr/bin/mail
```

Sticky Bit

The sticky bit is a permission bit that protects the files within a directory.

If the directory has the sticky bit set, a file can be deleted only by the owner of the file, the owner of the directory, or by root.

Example: This special permission prevents a user from deleting other users' files from public directories such as /tmp:

```
drwxrwxrwt 7 root sys 400 Sep 3 13:37 tmp
```

`--S------`	setuid is set, but user (owner) execute permission is not set
`--s------`	setuid and user execute persmission are both set
`-----S---`	setgid is set, but group execute permission is not set
`-----s---`	setgid and group execute permission are both set
`--------T`	sticky bit is set, but other execute permission is not set
`--------t`	sticky bit and other execute permission are both set

Note: Use setuid, and setgid with caution, incorrect use of setuid, and setgid can pose a security risk.

Notes

- Based on these articles:
  - Understanding and Setting UNIX File Permissions
  - Linux File Permissions, chmod, & umask
  - How to use SETUID SETGID and Stickybit Permissions
  - Special File Permissions (setuid, setgid and Sticky Bit)
- Converted by Wesley Hill

You can modify and improve this cheat sheet here