

Checking file flags and extended attributes in macOS

Katsutoshi Seki | Published: August 9, 2026 | Source: <https://sekika.github.io/2026/08/09/mac-ls/>

The macOS `ls` command has several BSD-specific options that are not available in the Linux `ls` command (GNU coreutils). Among them, the `-O` and `-@` options are particularly useful when troubleshooting problems such as being unable to rename a file or getting an error when running `chmod`.

1. Example

Run:

```
ls -lO@
```

If you see something like:

```
-r-----@ 1 seki staff uchg 140425 report.pdf
```

you can tell two things:

- The `@` indicates that the file has extended attributes.
- The `uchg` flag indicates that the file has the user immutable flag set.

2. The `-O` option

The `-O` option displays the file's file flags. For example:

```
$ ls -lO
-r----- 1 seki staff uchg 140425 Jul 14 10:32 report.pdf
```

The `uchg` shown here stands for **user immutable flag**. When this flag is set, even the owner of the file is prevented from modifying its contents, renaming it, deleting it, or changing its permissions.

Some of the commonly encountered flags are:

Flag	Meaning
<code>uchg</code>	User immutable
<code>uappnd</code>	User append-only
<code>hidden</code>	Hide the file in Finder
<code>schg</code>	System immutable
<code>sappnd</code>	System append-only

For more details, see Apple's documentation on [BSD File Flags](#).

File flags can be modified with the `chflags` command. For example, to set the `uchg` flag:

```
chflags uchg filename
```

and to remove it:

```
chflags nouchg filename
```

When `uchg` is set, commands such as `chmod`, `mv`, and `rm` may fail even when run by the file's owner, producing errors such as `Operation not permitted` or `Permission denied`.

3. The `-@` option

The `-@` option displays the **extended attributes** (`xattrs`) associated with a file. For example:

```
$ ls -l@
-rw-r--r--@ 1 seki staff 140425 Jul 14 10:32 report.pdf
  com.apple.lastuseddate#PS   16
  com.google.drivefs.item-id  33
```

In this example, the file has the following two extended attributes:

- `com.apple.lastuseddate#PS`: Information related to the file's last-used date
- `com.google.drivefs.item-id`: A file identifier managed by Google Drive for Desktop

To inspect the contents of extended attributes in more detail, run:

```
xattr -l filename
```

An individual extended attribute can be removed with:

```
xattr -d attribute_name filename
```

File flags and extended attributes are easy to confuse, but they serve different purposes:

- **File flags** are special attributes managed by the filesystem. They can affect how a file behaves, for example by making it immutable or append-only.
- **Extended attributes** are additional metadata attached to a file and are used by applications, Finder, and other parts of macOS.

4. The relationship between `uchg`, `schg`, and SIP

macOS has two types of file flags for protecting files: **user-level** and **system-level** flags.

Flag	Name	Who can set/remove it	Purpose
<code>uchg</code>	User immutable	Owner or root	Prevents modification by the user
<code>uappnd</code>	User append-only	Owner or root	Allows only appending by the user
<code>schg</code>	System immutable	root (with restrictions)	Prevents changes at the system level
<code>sappnd</code>	System append-only	root (with restrictions)	Allows only system-level appending

The `schg` flag is the **system immutable flag** and is stronger than `uchg`. On systems such as FreeBSD, a user with root privileges can set and remove it. On macOS, however, it is affected by **SIP (System Integrity Protection)**, so even root cannot necessarily remove it under normal circumstances.

[SIP \(System Integrity Protection\)](#) was introduced in OS X 10.11 El Capitan as a security mechanism to protect system files and important directories. When SIP is enabled, changes to locations such as:

- /System
 - /usr (except /usr/local)
 - /bin
 - /sbin
 - certain parts of /Library
- are restricted even for users with root privileges.

SIP also provides protection involving certain system file flags, such as schg, meaning that root privileges alone may not be sufficient to modify them.

The roles can be summarized as follows:

Feature	Scope	Main purpose
uchg	Individual files	Prevents users from modifying files
schg	Individual files	Prevents changes at the system level
SIP	The operating system	Protects system areas and critical files

In other words, uchg and schg are **file flags at the filesystem level**, whereas SIP is a **system-wide security mechanism in macOS**.

For the kinds of problems ordinary users encounter — such as being unable to rename a file or getting an error from chmod — the cause is often uchg, which can be checked with:

`ls -l0`

On the other hand, when you encounter problems modifying system files, the restriction is more often caused by SIP than by schg.

So, if a file on your Mac suddenly refuses to be renamed, deleted, or have its permissions changed, `ls -l0@` is a useful first diagnostic command: it lets you check both **file flags** and **extended attributes** in one shot.