WIKIPEDIA

Hidden file and hidden directory

In computing, a **hidden folder** (sometimes **hidden directory**) or **hidden file** is a folder or file which filesystem utilities do not display by default when showing a directory listing. They are commonly used for storing user preferences or preserving the state of a utility, and are frequently created implicitly by using various utilities. They are not a security mechanism because access is not restricted - usually the intent is simply to not "clutter" the display of the contents of a directory listing with files the user did not directly create.[1][2][3][4]

Contents

Unix and Unix-like environments
- Android
- GNOME
- macOS

DOS and MS Windows

References

External links

Unix and Unix-like environments

In Unix-like operating systems, any file or folder that starts with a dot character (for example, `/home/user/.config`), commonly called a **dot file** or **dotfile**, is to be treated as hidden – that is, the `ls` command does not display them unless the `-a` or `-A` flags (`ls -a` or `ls -A`) are used.[5] In most command-line shells, wildcards will not match files whose names start with `.` unless the wildcard itself starts with an explicit `.`.

A convention arose of using dotfiles in the user's home directory to store per-user configuration or informational text. Early uses of this were the well-known dotfiles `.profile`, `.login`, and `.cshrc`, which are configuration files for the Bourne shell and C shell and shells compatible with them, and `.plan` and `.project`, both used by the `finger` and `name` commands.[6]

Many applications, from bash to desktop environments such as GNOME now store their per-user configuration this way, but the Unix/Linux freedesktop.org *XDG Base Directory Specification* aims to migrate user config files from individual dotfiles in `$HOME` to non-hidden files in the hidden directory `$HOME/.config`.[7]

Android

The Android operating system uses empty `.nomedia` files to tell smartphone apps not to display or include the contents of the folder. This prevents digital photos and digital music files from being shown in picture galleries or played in MP3 player apps. This is useful to prevent downloaded voicemail files from playing between the songs in a playlist, and to keep personal photos private while still allowing those in other folders to be shared in person with friends, family, and colleagues. The `.nomedia` file has no effect on the filesystem or even the operating system, but instead depends entirely on each individual app to respect the presence of the different files.

GNOME

In the GNOME desktop environment (as well as all programs written using GLib[8]), filenames listed in a file named `.hidden` in each directory are also excluded from display. In GNOME's file manager, the keyboard shortcut Ctrl + H enables the display of both kinds of hidden files.

macOS

In addition to the "dotfile" behaviour, files with the "Invisible" attribute are hidden in Finder, although not in `ls`. The "Invisible" attribute can be set or cleared using the `SetFile` command; for example, invoking `SetFile -a V jimbo` will hide the file "jimbo".[9] Starting in Mac OS X Snow Leopard, the `chflags` command can also be used; for example, `chflags hidden jimbo` will hide the file "jimbo".[10]

DOS and MS Windows

In DOS systems, file directory entries include a Hidden file attribute which is manipulated using the `attrib` command. Using the command line command `dir /ah` displays the files with the Hidden attribute. In addition, there is a System file attribute that can be set on a file, which also causes the file to be hidden in directory listings. Use the command line command `dir /as` to display the files with the System attribute.

Under Windows Explorer, Hidden files and directories are, by default, not displayed - though they are still accessible by entering the full path into the explorer address bar. System files *are* displayed, unless they are also hidden. There are two options that enable the display of hidden files. The main 'Hidden files and folders' option can be used to turn on the display of hidden files but this won't, on its own, display hidden *system* files. A second option, 'Hide protected operating system files' additionally needs to be turned off in order for hidden system files to be shown. Hidden files are displayed with a slight transparency, so even when they are visible they are visually delineated from non-hidden files.

Under Windows Explorer, the content of a directory can also be hidden just by appending a pre-defined CLSID[11] to the end of the folder name. The directory is still visible, but its content becomes one of the Windows Special Folders.[12] However, the real content of this directory can still be seen using the CLI command `dir`.

References

1. "What is a hidden file?" (http://windows.microsoft.com/en-nz/windows/what-is-hidden-file#1TC=windows-7), Microsoft.com
2. "Configuring X: What are all those dotfiles for anyway?" (http://new.linuxfocus.org/English/March1998/article24.html). Linux Focus. March 1998. Retrieved 2013-09-08.
3. "Sample .bashrc and .bash_profile Files" (http://tldp.org/LDP/abs/html/sample-bashrc.html). Linux Documentation Project.
4. Subodh Soni (2001-12-01). "Understanding Linux configuration files" (https://www.ibm.com/developerworks/library/l-config/index.html). *IBM Developer*. Retrieved 2019-04-26.
5. "ls(1) - Linux man page" (https://linux.die.net/man/1/ls). linux.die.net. Retrieved 2020-08-23.
6. One user could lookup another by using the command along with the username (and hostname if not on the local host), and the finger service would respond with the other user's current status, and the contents of the `.plan` and `.project` files in that user's `$HOME` folder.
7. Bastian, Waldo; Lortie, Ryan; Poettering, Lennart (November 24, 2010). "XDG Base Directory Specification" (http://standards.freedesktop.org/basedir-spec/basedir-spec-latest.html). Retrieved June 4, 2014.
8. "GLib commit: Support for .hidden files" (https://git.gnome.org/browse/glib/commit/?id=510ba9b4efe1813e24c6dfa7405c3547bf9efdd7). Retrieved 2013-08-07.
9. `SetFile(1) (https://www.unix.com/man-page/mojave/1/SetFile)` – Darwin and macOS General Commands Manual

10. `chflags(1)` (https://www.unix.com/man-page/mojave/1/chflags) – Darwin and macOS General Commands Manual
11. "Canonical Names of Control Panel Items - Win32 apps" (https://docs.microsoft.com/en-us/windows/win32/shell/controlpanel-canonical-names?redirectedfrom=MSDN). *Microsoft*.
12. "The Secret BEHIND the Windows 7 GodMode" (https://www.thewindowsclub.com/the-secret-behind-the-windows-7-godmode). *The Windows Club*. 5 January 2010.

External links

- Bellevue Linux Users Group: Hidden files in Unix-like operating systems (https://web.archive.org/web/20071017224738/http://bellevuelinux.org/hidden_file.html) at the Wayback Machine (archived October 17, 2007)
- Computer Hope: Microsoft DOS `attrib` command (http://www.computerhope.com/attribhl.htm)
- .NOMEDIA file (https://nomedia.net)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Hidden_file_and_hidden_directory&oldid=988880696"

This page was last edited on 15 November 2020, at 20:22 (UTC).