Kaya Nelson

Micheal Landreth

CST – 221

6/11/2024

**Table 1: File Systems**

|  |  |
| --- | --- |
| **Directory** | **Purpose** |
| / | The root directory, signified by the forward slash /, is the top-level directory in a file system hierarchy. It stores all the essential system software and files required for the operating system to boot and function properly. The root directory contains subdirectories that hold various system and user files and programs. |
| /bin | The /bin directory is the location in the file system hierarchy that stores essential binary executable files. These are the basic command-line programs and utilities that are necessary for the operating system to function, such as the shell, file management tools, and other core system commands that can be executed from the terminal or command prompt. |
| /dev | The /dev directory is a special directory in the file system that contains device files representing the hardware devices connected to the system. These device files allow the operating system to interact with and manage the hardware components. The /dev directory is essential for the proper functioning of the system, as without it, the operating system would not be able to communicate with the hardware and the system would likely cease to function properly. |
| /etc | The /etc directory is the location in the file system hierarchy that stores the system's global configuration files. These are the files that contain settings and preferences that affect the entire system and apply to all users. The files in the /etc directory are essential for the proper setup and operation of the operating system and its various components. Modifying these configuration files can have a significant impact on the overall system functionality. |
| /lib | The /lib directory is the location in the file system hierarchy that stores the system's shared libraries and kernel modules. Shared libraries are code packages that provide common functionality used by multiple programs. Kernel modules are pieces of code that can be dynamically loaded and unloaded to extend the functionality of the operating system kernel. The /lib directory is essential for the proper operation of the system, as many programs and services rely on the libraries and modules found in this directory. |
| /boot | The /boot directory is the location in the file system hierarchy that stores the static boot loader files, including the kernel image and any initial RAM disk images. These files are essential for the boot process, as they contain the core components required to load the operating system into memory and begin the boot sequence. The /boot directory is a critical part of the file system, as without these boot-related files, the system would be unable to successfully boot and start up. |
| /home | The /home directory is the location in the file system hierarchy that stores the individual home directories for each user of the system. Each user account on the system has their own subdirectory within the /home directory, which is where all of that user's personal files, configurations, and data are stored. The /home directory is an important part of the file system, as it provides a dedicated and private space for each user to store their own documents, settings, and other personal information. Without the /home directory, users would not have a designated place to keep their own data separate from the system-wide files. |
| /mnt | The /mnt directory is a standard location in the file system hierarchy used for temporarily mounting additional file systems, such as network-attached storage, removable media, or other remote file shares. When a new file system needs to be accessed, it is typically mounted under the /mnt directory, allowing the contents of that file system to be accessed and interacted with as part of the overall file system structure. The /mnt directory provides a convenient and standardized way for the operating system and users to manage the temporary attachment of additional file systems to the main file system. |
| /proc | The /proc directory is a special virtual file system in the Linux file system hierarchy. It is not a physical storage location, but rather a interface provided by the kernel to allow processes to access and exchange data with the operating system kernel. The /proc file system exposes various kernel parameters, process information, and other system-level data that can be read from or written to by running processes. This allows programs to dynamically interact with and monitor the state of the kernel and the overall system. The /proc virtual file system is an essential component that facilitates communication between running processes and the core operating system kernel. |
| /tmp | The /tmp directory is the standard location in the Linux file system hierarchy where applications can store temporary files. This directory is intended for the storage of transient data that does not need to persist beyond the lifetime of the running application. Examples of files stored in /tmp include internet cookies, cached data, and other temporary artifacts created by programs during their execution. The contents of the /tmp directory are typically cleared on system reboot or when the system administrator deems it necessary to clear out old temporary files. Keeping sensitive data out of the /tmp directory is important, as it is a shared resource that may not provide the same level of privacy and security as a user's own home directory. |
| /usr | The /usr directory is a major part of the Linux file system hierarchy that contains user-level applications, utilities, and other resources. Unlike the /bin and /sbin directories that store system-level binaries, the /usr directory is where user-installed or user-specific programs are typically located. This includes things like desktop applications, programming tools, libraries, and other user-facing software. The /usr directory provides a central location for users to access the various applications and utilities they need to interact with and utilize the capabilities of the overall system. It is an essential part of the file system structure that enables users to customize and extend the functionality of their computing environment. |
| /var | The /var directory is the location in the Linux file system hierarchy reserved for storing variable data files. This includes system logs, databases, website content, email spools, and other types of data that are expected to change frequently over time. The /var directory provides a dedicated space for applications and services to store their dynamic, mutable data, keeping it separate from the more static configuration and program files found elsewhere in the file system. Maintaining a distinct /var directory helps to organize the file system and ensures that growing or changing data does not inadvertently fill up critical system directories. The contents of /var are an essential part of the overall functionality and state of the running system. |
| /sbin | The /sbin directory is the location in the Linux file system hierarchy that stores system administration binaries and commands that require superuser or root privileges to execute. These are essential low-level system utilities and tools that are used for tasks like system startup, network configuration, filesystem maintenance, and other core administrative functions. The /sbin directory is distinct from the /bin directory, which contains general user-level commands accessible to all accounts. Placing the critical system administration tools in the /sbin directory helps to restrict their access and usage to only authorized superuser accounts, maintaining the overall security and integrity of the system. |
| /kernel | The /usr/src directory is a location in the Linux file system hierarchy that is not typically installed or populated by default. This directory is intended to store the full source code for the operating system kernel, which can be customized and compiled by programmers as needed. The kernel source code in /usr/src allows advanced users and developers to modify, extend, or optimize the core operating system to suit their specific requirements. While the pre-compiled kernel binaries are found in other directories like /boot, the /usr/src directory provides access to the complete kernel source for those who wish to build a custom kernel image. This level of configurability is an important aspect of the open-source nature of Linux operating systems. |

**Root Directory:**On macOS (the operating system for your MacBook), the directory structure may not fully mirror the standard Linux file system hierarchy. The /root directory is typically used in Linux as the home directory for the superuser or "root" account, but on macOS this directory may have a different purpose or not exist at all. The specific directories found under /root can vary depending on the macOS installation and how the system has been configured.

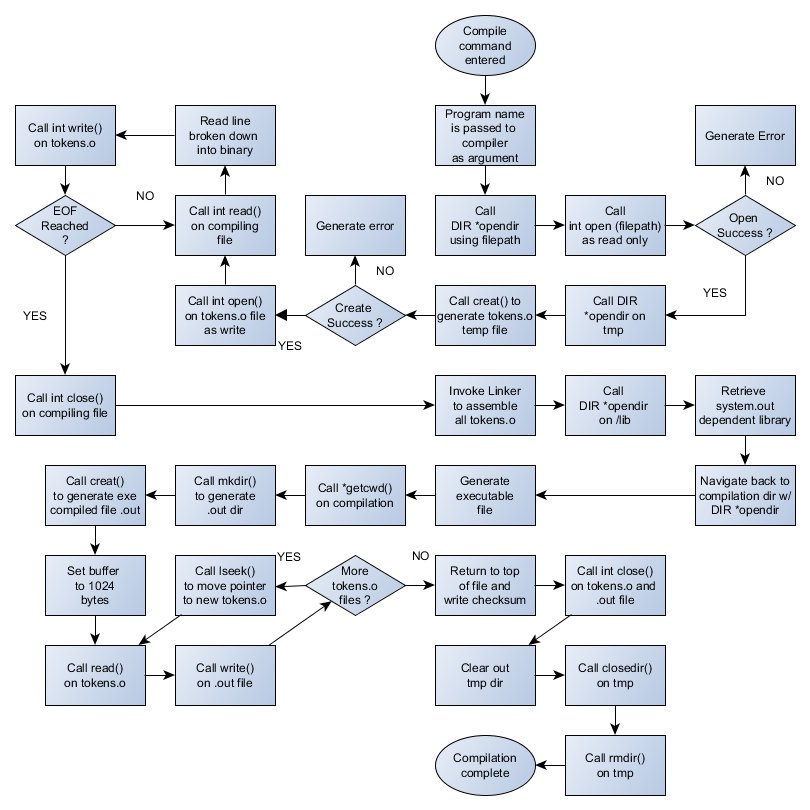
**C Compiler Flow Chart:**Figure 1 below shows a high-level flow chart diagramming the process of compiling a C program. The main steps are:

**A.** Parsing the user's C code and generating temporary token files. This involves breaking down the source code into its lexical components.

**B.** Linking the temporary token files with the required system libraries to produce the final executable program. This step combines the parsed code with the necessary system functionality to create the runnable binary.

Of course, this flow chart simplifies the full complexity of the C compilation process. Things like error checking, optimization, and other advanced compiler features are not shown here but would be covered in more detail in a computer science course. The chart provides a basic overview of the two main stages involved in transforming source code into an executable program.

**Flowchart for simulated C Compiler**



**References**

*LinuxFilesystemTreeOverview - Community Help Wiki*. (n.d.). <https://help.ubuntu.com/community/LinuxFilesystemTreeOverview>

*What’s the “/sys” directory for?* (n.d.). Ask Ubuntu. <https://askubuntu.com/questions/720471/whats-the-sys-directory-for>

Okoi, D., & Okoi, D. (2017, February 27). *What are snaps? and how are they important?* GeeksMint: Computers, How-to’s, Internet, Tips and Tricks. <https://www.geeksmint.com/what-are-ubuntu-snaps-and-how-are-they-important/>

Hoffman, C. (2016, September 21). The Linux directory structure, explained. *How-To Geek*. <https://www.howtogeek.com/117435/htg-explains-the-linux-directory-structure-explained/>

*What is the difference between /opt and /usr/local?* (n.d.). Unix & Linux Stack Exchange. <https://unix.stackexchange.com/questions/11544/what-is-the-difference-between-opt-and-usr-local>

GeeksforGeeks. (2023, April 14). *Compiling a C program behind the scenes*. GeeksforGeeks. <https://www.geeksforgeeks.org/compiling-a-c-program-behind-the-scenes/>