Linux All-in-One For Dummies

From Linux All-in-One For Dummies, 5th Edition by Emmett Dulaney

Linux can fulfill almost any need you have for the operating system on a desktop computer, but you must be able to tell it what you want to do in a way that it understands. First, you have to get Linux installed. Then, you need to know about its important directories and common commands, how to work with file permissions, how to check network configuration and connectivity, and how to interact with the bash shell.

Linux Installation Checklist

Every Linux distribution differs slightly in the interface used to install the operating system on your desktop, the order of the information it requests, and the tools it uses to simplify the installation process. Regardless of the distribution, however, you must follow these four major steps to install Linux:

1. **If you want to keep your existing operating system, resize the disk partition by using a partitioning tool or get a second hard drive.**
2. **Burn DVDs or format flash drives for your distribution and boot the PC from the DVD or flash drive.**
3. **Go through the graphical installation steps.**
4. **Configure other hardware when the system first boots.**

Linux Commands to Check the Network

Many Linux distributions provide a number of tools to simplify network configuration and connectivity verification. The following tools work with every distribution and allow you to administer your network from the command line:

- `ping`: Checks network connectivity
- `ifconfig`: Displays the configuration for a network interface
- `traceroute`: Shows the path taken to reach a host
- `route`: Displays the routing table and/or lets you configure it
- `arp`: Shows the address resolution table and/or lets you configure it
- `netstat`: Displays the status of the network

Working in Linux with the bash Shell

The bash shell serves as the mediator between the user and the Linux kernel, with bash being the most common shell in use today. To interact with the bash shell, you need to understand how the pipe works (allowing the output of one command to be the input of the next), how to use redirection, and some basic commands and environment variables:

- **Pipe:**
  - *command1* | *command2*
- **Redirections:**
  - *command* > *file*: output goes to `file`
  - *command* < *file*: input from `file`

- *command* `>>` `file`: append to `file`
- *command2*`>` `file`: errors go to `file`

- **Commands:**
  - `alias`: Defines a shortcut for a long command
  - `apropos`: Searches the manpages for keywords
  - `history`: Displays the most recent commands
  - `locate`: Finds files
  - `whereis`: Finds executable files for a command
  - `which`: Shows the full pathname for a command
  - `man`: Displays online help
  - `printenv`: Displays the environment variables
- **Environment variables:**
  - `HOME`: User's home directory
  - `PATH`: Directories to search for commands
  - `TERM`: Name of a terminal type

Linux File System Basics

To interact with the Linux file system, you must understand some basics: its key directories, some of the most commonly used commands, and the permissions you can assign to files.

- **Key directories in the file system:**
  - `*/`: Root directory (base of file system)
  - `/bin`: Executable programs
  - `/boot`: Linux kernel and boot loader
  - `/dev`: Special device files
  - `/etc`: System configuration files
  - `/home`: Home directories of all users
  - `/lib`: Library files for programs
  - `/media`: Mount points for CD-ROM and other media
  - `/root`: Home directory of the `root` user
  - `*/sbin`: System administration commands
  - `/srv`: Data for services such as Web and FTP
  - `*/tmp`: Temporary directory
  - `/usr`: Many of the important administration programs
  - `/var`: Various system files, such as logs
- **Common commands:**
  - `cat`: Copies a file to the standard output
  - `cd`: Changes the current directory
  - `chmod`: Changes file permissions
  - `chown`: Changes file ownerships
  - `cp`: Copies files
  - `dd`: Copies blocks of data
  - `df`: Reports disk space usage by device and available space
  - `diff`: Compares two text files
  - `du`: Reports disk space usage by directory

- `file`: Displays the type of data in a file
- `find`: Finds files based on specified criteria
- `grep`: Searches for text in a file
- `ln`: Links a filename to an alias name
- `ls`: Displays the contents of a directory
- `mkdir`: Creates a directory
- `more`: Displays a text file, one page at a time
- `mount`: Mounts a file system
- `mv`: Renames or moves a file
- `pwd`: Displays the current directory
- `rm`: Deletes files
- `rmdir`: Deletes directories
- `sort`: Sorts lines in a text file
- `split`: Splits a file into smaller parts
- `umount`: Unmounts a file system
- `wc`: Counts the words and lines in a file

- **File permissions:**
  - `rwxrwxrwx`: Three sets of `rwx`. The leftmost set pertains to the owner, the middle set is for the group, and the rightmost set is for others; `rwx` stands for read (`r`), write (`w`), execute (`x`); the dash (`-`) means no permission.
  - `rwx------`: Only the owner can read, write, and execute.
  - `rw-r--r--`: Everyone can read, and the owner can also write.
  - `rw-------`: Only the owner can read and write.
  - `r--r--r--`: Everyone can read.
  - Permissions can also be expressed numerically, where read (`r`) is equal to 4, write (`w`) is equal to 2, execute (`x`) is equal to 1, and no permission is equal to 0. Therefore, `rwxrwxrwx` is equal to 777, `rwx------` is equal to 700, `rw-r--r–` is equal to 644, `rw-------` is equal to 600, and `r--r--r--` is equal to 444.