



LiNUX

Linux Miscellaneous Basics including VIM Editor

Day 01, Session 1.4

Important Linux Commands

This section gives insight into the most important commands of your Linux system. Along with the individual commands, parameters are listed and, where appropriate, a typical sample application is introduced. To learn more about the various commands, it is usually possible to get additional information with the **man** program followed by the name of the command,

for example, **man ls**.

In these *manual pages*, move up and down with **PgUp** and **PgDn** and move between the beginning and the end of a document with **Home** and **End**. End this viewing mode by pressing **Q**.

Learn more about the **man** command itself with **man**

Please Remember

- The actual command is always printed as **command**. Without this, nothing can function.
- Options without which the respective program cannot function are printed in italics.
- Further details, like file names, which must be passed to a command for correct functioning, are written in the Courier font.
- Specifications or parameters that are not required are placed in [brackets].
- Adjust possible specifications to your needs. It makes no sense to write **ls file(s)**, if no file named file(s) actually exists. You can usually combine several parameters,

for example, by writing **ls -la** instead of **ls -l -a**.

ls [option(s)] [file(s)]

If you run **ls** without any additional parameters, the program will list the contents of the current directory in short form.

-l

detailed list

-a

displays hidden files

cp [option(s)] sourcefile targetfile

Copies sourcefile to targetfile.

-i

Waits for confirmation, if necessary, before an existing targetfile is overwritten

-r

mv [option(s)] sourcefile targetfile

Copies sourcefile to targetfile then deletes the original sourcefile.

-b

Creates a backup copy of the sourcefile before moving

-i

Waits for confirmation, if necessary, before an existing targetfile is overwritten

rm [option(s)] file(s)

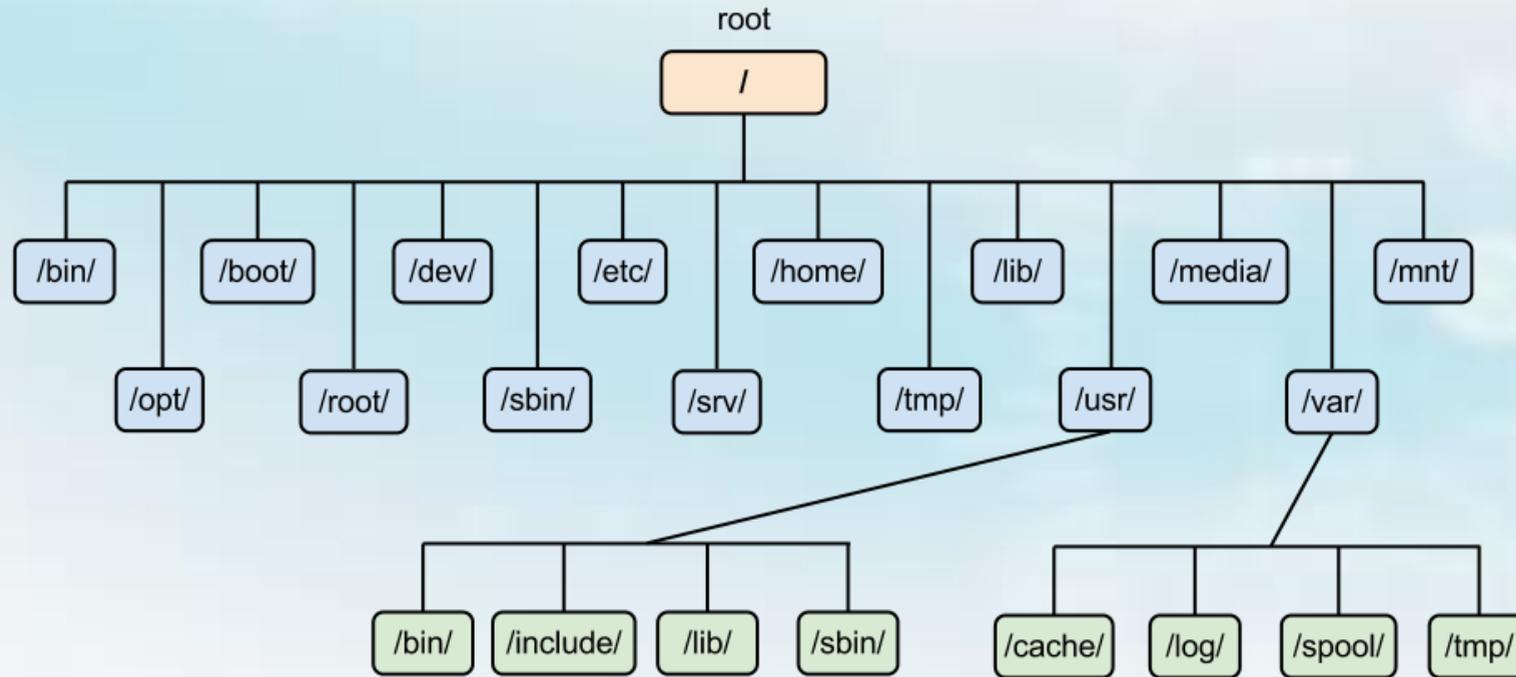
Removes the specified files from the file system. Directories are not removed by **rm** unless the option -r is used.

-r

Deletes any existing subdirectories

-i

Waits for confirmation before deleting each file.



cd [options(s)] [directory]

Changes the current directory. **cd** without any parameters changes to the user's home directory.

mkdir [option(s)] directoryname

Creates a new directory.

rmdir [option(s)] directoryname

Deletes the specified directory, provided it is already empty.

passwd [option(s)] [username]

Users may change their own passwords at any time using this command. Furthermore, the administrator root can use the command to change the password of any user on the system.

su [option(s)] [username]

The **su** command makes it possible to log in under a different user name from a running session. When using the command without specifying a user name, you will be prompted for the root password. Specify a user name and the corresponding password to use the environment of the respective user. The password is not required from root, as root is authorized to assume the identity of any user.

reboot [option(s)]

System performs an immediate reboot.

clear

This command cleans up the visible area of the console. It has no options.

top [options(s)]

top provides a quick overview of the currently running processes. Press **H** to access a page that briefly explains the main options to customize the program.

ps [option(s)] [process ID]

If run without any options, this command displays a table of all *your own* programs or processes — those you started. The options for this command are not preceded by hyphen.

aux

Displays a detailed list of all processes, independent of the owner.

kill [option(s)] process ID

Unfortunately, sometimes a program cannot be terminated in the normal way. However, in most cases, you should still be able to stop such a runaway program by executing the **kill** command, specifying the respective process ID (see **top** and **ps**).

kill sends a *TERM* signal that instructs the program to shut itself down. If this does not help, the following parameter can be used:

-9

Sends a *KILL* signal instead of a *TERM* signal, with which the process really is *annihilated* by the operating system. This brings the specific processes to an end in almost all cases

Vim is a highly configurable text editor for efficiently creating and changing any kind of text. It is included as "vi" with most UNIX systems and with Apple OS X.

Vim is rock stable and is continuously being developed to become even better. Among its features are:

- persistent, multi-level undo tree
- extensive plugin system
- support for hundreds of programming languages and file formats
- powerful search and replace
- integrates with many tools



Insert Mode

The Insert mode lets you insert text in a document. The shortcut is: "i" (insert text where the cursor is) or "o" (insert text at the beginning of the following line).

Visual Mode

The visual mode permits the user to select the text like you would do with a mouse, but using the keyboard instead of the mouse. Useful to copy several lines of text for example. The shortcut is: "V".

Command Mode

Let's now speak about the command mode, a command begins with the symbol ":"

When you are in another mode you can use the **escape key** (sometimes you'll need to hit it twice) to come back to command mode at any time.

Vim - text editor

To start editing the file content, enter:

:i [enter]

The word **--insert--** will appear at the bottom of the editor window to show that you are in insert mode now

Now you can edit the file by navigating to the line that you want to change with the cursor keys and then start typing the text. When you are finished with editing, press the [esc] key to go back to the command mode.

To save the file and exit the editor, enter:

:x [enter]

In case you want to quit vim without saving the file, enter:

:q! [enter]

Vim Command Reference

save: :w

save and exit: :wq

exit: :q

force: ! (example :w! :q!)

copy: y

copy a line: yy

paste: p

cut: d

cut a line: dd



Any Questions ?