

Cheat Sheet Linux Commands

LINUX COMMANDS CHEAT SHEET	
SYSTEM <code>#uname -a</code> => Display linux system information <code>#uname -r</code> => Display kernel release information <code>#uptime</code> => Show how long the system has been running + load <code>#hostname</code> => Show system host name <code>#hostname -i</code> => Display the IP address of the host <code>#last reboot</code> => Show system reboot history <code>#date</code> => Show the current date and time <code>#cal</code> => Show this month calendar <code>#w</code> => Display who is online <code>#whoami</code> => Who you are logged in as <code>#finger user</code> => Display information about user	FILE PERMISSION RELATED <code>#chmod octal file-name</code> => Change the permissions of file to octal Example <code>#chmod 777 /data/test.c</code> => Set rwx permission for owner,group,world <code>#chmod 755 /data/test.c</code> => Set rwx permission for owner,rx for group and world <code>#chown owner-user file</code> => Change owner of the file <code>#chown owner-user:owner-group file-name</code> => Change owner and group owner of the file <code>#chown owner-user:owner-group directory</code> => Change owner and group owner of the directory
HARDWARE <code>#dmesg</code> => Detected hardware and boot messages <code>#cat /proc/cpuinfo</code> => CPU model <code>#cat /proc/meminfo</code> => Hardware memory <code>#cat /proc/interrupts</code> => Lists the number of interrupts per CPU per I/O device <code>#lshw</code> => Displays information on hardware configuration of the system <code>#lshw -b</code> => Displays block device related information in Linux <code>#free -m</code> => Used and free memory (-m for MB) <code>#lspci -tv</code> => Show PCI devices <code>#lsusb -tv</code> => Show USB devices <code>#lsmcode</code> => Show hardware info from the BIOS <code>#hdparm -i /dev/sda</code> => Show info about disk sda <code>#hdparm -T /dev/sda</code> => Do a read speed test on disk sda <code>#badblocks -s /dev/sda</code> => Test for unreadable blocks on disk sda	NETWORK <code>#ifconfig -a</code> => Display all network ports and ip address <code>#ifconfig eth0</code> => Display specific ethernet port <code>#ethtool eth0</code> => Linux tool to show ethernet status <code>#mii-tool eth0</code> => Linux tool to show ethernet status <code>#ping host</code> => Send echo request to test connection <code>#whois domain</code> => Get who is information for domain <code>#dig domain</code> => Get DNS information for domain <code>#dig -x host</code> => Reverse lookup host <code>#host google.com</code> => Lookup DNS ip address for the name <code>#hostname -i</code> => Lookup local ip address <code>#wget file</code> => Download file <code>#netstat -tul</code> => List active connections to / from system
USERS <code>#id</code> => Show the active user id with login and group <code>#last</code> => Show last logins on the system <code>#who</code> => Show who is logged on the system <code>#groupadd admin</code> => Add group "admin" <code>#useradd -c "Sam Tomsh" -s /bin/bash -m -s /bin/bash</code> => Create user "sam" <code>#userdel sam</code> => Delete user sam <code>#adduser sam</code> => Add user "sam" <code>#usermod</code> => Modify user information	COMPRESSION / ARCHIVES <code>#tar cf home.tar home/</code> => Create tar named home.tar containing home/ <code>#tar xf file.tar</code> => Extract the files from file tar <code>#tar czf file.tar.gz files</code> => Create a tar with gzip compression <code>#gzip file</code> => Compress file and renames it to file.gz
FILE COMMANDS <code>#ls -al</code> => Display all information about files/ directories <code>#pwd</code> => Show the path of current directory <code>#mkdir directory-name</code> => Create a directory <code>#rm file-name</code> => Delete file <code>#rm -r directory-name</code> => Delete directory recursively <code>#rm -f file-name</code> => Forcefully remove file <code>#rm -rf directory-name</code> => Forcefully remove directory recursively <code>#cp file1 file2</code> => Copy file1 to file2 <code>#cp -r dir1 dir2</code> => Copy dir1 to dir2, create dir2 if it doesn't exist <code>#mv file1 file2</code> => Rename source to dest / move source to directory <code>#ln -s /path/to/file-name link-name</code> => Create symbolic link to file-name <code>#touch file</code> => Create or update file <code>#cat > file</code> => Place standard input into file <code>#more file</code> => Output contents of file <code>#head file</code> => Output first 10 lines of file <code>#tail file</code> => Output last 10 lines of file <code>#tail -f file</code> => Output contents of file as it grows starting with the last 10 lines <code>#pgp -c file</code> => Encrypt file <code>#pgp -d file.pgp</code> => Decrypt file <code>#wc</code> => Print the number of bytes, words, and lines in files <code>#xargs</code> => Execute command lines from standard input	INSTALL PACKAGE <code>#rpm -i pkgname.rpm</code> => Install rpm based package <code>#rpm -e pkgname</code> => Remove package INSTALL FROM SOURCE <code>#./configure</code> <code>#make</code> <code>#make install</code>
SEARCH <code>#grep pattern files</code> => Search for pattern in files <code>#grep -r pattern dir</code> => Search recursively for pattern in dir <code>#locate file</code> => Find all instances of file <code>#find /home/tom -name "index"</code> => Find files names that start with "index" <code>#find /home -size +10000k</code> => Find files larger than 10000k in /home	LOGIN (SSH AND TELNET) <code>#ssh user@host</code> => Connect to host as user <code>#ssh -p port user@host</code> => Connect to host using specific port <code>#telnet host</code> => Connect to the system using telnet port
PROCESS RELATED <code>#ps</code> => Display your currently active processes <code>#ps aux grep 'telnet'</code> => Find all process id related to telnet process <code>#mpmap</code> => Memory map of process <code>#top</code> => Display all running processes <code>#killpid</code> => Kill process with mentioned pid id <code>#killall proc</code> => Kill all processes named proc <code>#kill process-name</code> => Send signal to a process with its name <code>#fg</code> => Brings the most recent job to foreground <code>#bg</code> => Brings job n to the foreground <code>#fg n</code> => Brings job n to the foreground	FILE TRANSFER <code>scp</code> <code>#scp file.txt server2:/tmp</code> => Secure copy file.txt to remote host /tmp folder <code>rsync</code> <code>#rsync -a /home/apps /backup/</code> => Synchronize source to destination
DISK USAGE <code>#df -h</code> => Show free space on mounted filesystems <code>#df -i</code> => Show free inodes on mounted filesystems <code>#fdisk -l</code> => Show disks partitions sizes and types <code>#du -sh</code> => Display disk usage in human readable form <code>#du -sh</code> => Display total disk usage on the current directory	DIRECTORY TRAVERSE <code>#cd ..</code> => To go up one level of the directory tree <code>#cd</code> => Go to \$HOME directory <code>#cd /test</code> => Change to /test directory



MORE DETAILED : [HTTP://LINUXIDE.COM/GUIDE/LINUX-COMMAND-SHELF.HTML](http://linuxide.com/guide/linux-command-shelf.html)

Cheat sheet linux commands are an essential resource for both novice and experienced Linux users. Whether you're a system administrator, a developer, or simply someone who wants to navigate the Linux command line more efficiently, having a handy reference can save time and reduce frustration. This article will provide a comprehensive overview of some of the most important Linux commands, organized into categories for easy reference.

Getting Started with Linux Commands

Linux commands are executed in the terminal, which is a powerful tool for interacting with the operating system. Before diving into specific commands, it's essential to understand a few fundamental concepts.

Terminal Basics

- Opening the Terminal: You can usually find the terminal application in the applications menu. Alternatively, you can use keyboard shortcuts like ``Ctrl + Alt + T``.
- Command Structure: A command typically consists of the command itself followed by options and arguments. For example, ``ls -l /home/user`` where ``ls`` is the command, ``-l`` is an option, and ``/home/user`` is an argument.

Common Terminal Shortcuts

- ``Ctrl + C``: Cancel the current command.
- ``Ctrl + Z``: Suspend the current command.
- ``Ctrl + D``: Log out of the current session.
- ``Tab``: Autocomplete commands or file names.

File and Directory Management

Managing files and directories is one of the most common tasks performed in Linux. Here are some essential commands:

File Commands

1. ``ls``: List directory contents.
 - Usage: ``ls [options] [directory]``
 - Options:
 - ``-l``: Long listing format.
 - ``-a``: Show hidden files.
2. ``cp``: Copy files and directories.
 - Usage: ``cp [options] source destination``
 - Options:
 - ``-r``: Recursive copy for directories.
 - ``-i``: Prompt before overwrite.
3. ``mv``: Move or rename files and directories.

- Usage: ``mv [options] source destination``
 - Options:
 - ``-i``: Prompt before overwrite.
4. ``rm``: Remove files or directories.
 - Usage: ``rm [options] file``
 - Options:
 - ``-r``: Remove directories and their contents recursively.
 - ``-f``: Force removal without prompts.
 5. ``touch``: Create an empty file or update the timestamp of an existing file.
 - Usage: ``touch filename``

Directory Commands

1. ``mkdir``: Create a new directory.
 - Usage: ``mkdir [options] directory``
 - Options:
 - ``-p``: Create parent directories as needed.
2. ``rmdir``: Remove empty directories.
 - Usage: ``rmdir directory``
3. ``pwd``: Print the current working directory.
 - Usage: ``pwd``
4. ``cd``: Change the current directory.
 - Usage: ``cd [directory]``
 - Quick Commands:
 - ``cd ..``: Go up one directory.
 - ``cd ~``: Go to the home directory.

File Permission and Ownership

Understanding and managing file permissions is crucial for maintaining security in Linux.

Permissions Commands

1. ``chmod``: Change file permissions.
 - Usage: ``chmod [options] mode file``
 - Modes:
 - Numeric (e.g., ``755``, ``644``).
 - Symbolic (e.g., ``u+x``, ``g-w``).

2. `chown`: Change file owner and group.
 - Usage: `chown [options] owner:group file`
3. `chgrp`: Change group ownership.
 - Usage: `chgrp [options] group file`

System Information and Monitoring

Getting information about the system can help in troubleshooting and managing resources.

System Commands

1. `top`: Display dynamic real-time information about running processes.
 - Usage: `top`
2. `htop`: An improved version of `top` with a user-friendly interface.
 - Usage: `htop` (may need to install it first).
3. `df`: Display disk space usage.
 - Usage: `df [options]`
 - Options:
 - `-h`: Human-readable format.
4. `du`: Estimate file space usage.
 - Usage: `du [options] [directory]`
 - Options:
 - `-h`: Human-readable format.
 - `-s`: Summary of total space used.
5. `free`: Display memory usage.
 - Usage: `free [options]`
 - Options:
 - `-h`: Human-readable format.

Networking Commands

Networking is a critical aspect of Linux, and several commands can assist with network management.

Networking Commands

1. `ping`: Check the network connection to a host.

- Usage: ``ping [options] destination``
 - Options:
 - ``-c``: Specify the number of packets to send.
2. ``ifconfig``: Display or configure a network interface.
- Usage: ``ifconfig [interface] [options]``
3. ``ip``: A more advanced tool for network management.
- Usage: ``ip [options]``
 - Common subcommands:
 - ``ip addr``: Display IP addresses.
 - ``ip link``: Show network interfaces.
4. ``netstat``: Display network connections, routing tables, interface statistics.
- Usage: ``netstat [options]``
 - Options:
 - ``-t``: Show TCP connections.
 - ``-u``: Show UDP connections.
5. ``curl``: Transfer data from or to a server using various protocols.
- Usage: ``curl [options] [URL]``
 - Options:
 - ``-O``: Save the output to a file.

Package Management

Linux distributions use package managers to install, update, and remove software.

Debian-based Systems (e.g., Ubuntu)

1. ``apt-get``: Command-line interface for the Advanced Package Tool.
- Usage: ``sudo apt-get [command] [package]``
 - Common Commands:
 - ``install``: Install a package.
 - ``remove``: Remove a package.
 - ``update``: Update the list of available packages.
 - ``upgrade``: Upgrade installed packages.

Red Hat-based Systems (e.g., CentOS)

1. ``yum``: Package manager for RPM-based distributions.
- Usage: ``sudo yum [command] [package]``
 - Common Commands:

- ``install``: Install a package.
- ``remove``: Remove a package.
- ``update``: Update installed packages.

Searching and Finding Files

Finding files and searching for content in files is a common task in Linux.

Search Commands

1. ``find``: Search for files in a directory hierarchy.
 - Usage: ``find [path] [options] [expression]``
2. ``grep``: Search for patterns within files.
 - Usage: ``grep [options] pattern [file]``
 - Options:
 - ``-r``: Recursive search.
 - ``-i``: Ignore case.
3. ``locate``: Find files by name using a database.
 - Usage: ``locate filename``

Text Processing Commands

Linux provides a variety of commands for processing and manipulating text files.

Text Processing Commands

1. ``cat``: Concatenate and display files.
 - Usage: ``cat [options] [file]``
 - Options:
 - ``-n``: Number the output lines.
2. ``less``: View file contents one page at a time.
 - Usage: ``less [file]``
3. ``head``: Output the first part of files.
 - Usage: ``head [options] [file]``
 - Options:
 - ``-n``: Specify the number of lines.
4. ``tail``: Output the last part of files.

- Usage: ``tail [options] [file]``
- Options:
- ``-f``: Follow the file as it grows.

Conclusion

This cheat sheet linux commands serves as a quick reference to some of the most commonly used commands in Linux. While this guide covers a wide range of commands, Linux offers a vast array of tools that can be used for various tasks. Users are encouraged to explore the ``man`` command (manual) for more detailed information on each command. Remember, practice makes perfect; the more you use these commands, the more proficient you'll become in navigating the Linux environment.

Frequently Asked Questions

What are the most essential Linux commands every beginner should know?

Some essential commands include `'ls'` for listing files, `'cd'` for changing directories, `'cp'` for copying files, `'mv'` for moving files, `'rm'` for removing files, and `'man'` for accessing the manual.

How can I view the contents of a file in Linux?

You can use commands like `'cat'`, `'less'`, or `'more'` to view the contents of a file. For example, `'cat filename.txt'` will display the entire file content in the terminal.

What is the function of the 'grep' command?

`'grep'` is used to search for specific patterns within files. For example, `'grep 'text' filename.txt'` will search for the word `'text'` in the specified file.

How do I check disk usage in Linux?

You can use the `'df'` command to check disk space usage. For example, `'df -h'` displays disk space in a human-readable format.

What does the 'chmod' command do?

`'chmod'` is used to change the permissions of files or directories. For instance, `'chmod 755 filename'` sets the permissions to read, write, and execute for the owner, and read and execute for the group and others.

How can I find files in Linux?

You can use the 'find' command to search for files. For example, 'find /path/to/search -name filename.txt' will look for 'filename.txt' in the specified path.

What is the difference between 'sudo' and 'su'?

'sudo' allows a permitted user to execute a command as the superuser or another user, while 'su' switches the current user to another user, typically the root user.

How can I list all running processes in Linux?

You can list all running processes using the 'ps' command or 'top' for a dynamic view. For instance, 'ps aux' shows all running processes with details.

What command is used to update package lists in Debian-based systems?

In Debian-based systems like Ubuntu, you can update package lists by using the command 'sudo apt update'.

How can I create a new directory in Linux?

You can create a new directory using the 'mkdir' command. For example, 'mkdir new_directory' will create a directory named 'new_directory'.

Find other PDF article:

<https://soc.up.edu.ph/27-proof/files?dataid=qRr20-9818&title=heart-to-heart-with-mallory.pdf>

Cheat Sheet Linux Commands

Cheat Engine :: View topic - error in Lazarus

Jan 7, 2024 · Cheat Engine :: View topic - error in Lazarus

Cheat Engine :: View topic - Pointer scan

Mar 23, 2025 · Cheat Engine :: View topic - Pointer scan

Cheat Engine :: View topic - Bluestacks Help, Please

Apr 27, 2025 · Discussion forum for Cheat Engine users seeking assistance with Bluestacks.

Lua Script Cheat Table -- The Best Way - Cheat Engine

Mar 23, 2025 · Your 'Lua Script : Cheat Table' got so much lines of codes ? this is not a problem anymore ! The better way is to load '.lua' files directly inside the Cheat Table, there is an exemple :

Cheat Engine :: View topic - Speedhack

Apr 8, 2024 · Cheat Engine :: View topic - Speedhack

Cheat Engine :: View topic - error in Lazarus

Jan 7, 2024 · Cheat Engine :: View topic - error in Lazarus

Cheat Engine :: View topic - Pointer scan

Mar 23, 2025 · Cheat Engine :: View topic - Pointer scan

Cheat Engine :: View topic - Bluestacks Help, Please

Apr 27, 2025 · Discussion forum for Cheat Engine users seeking assistance with Bluestacks.

Lua Script Cheat Table -- The Best Way - Cheat Engine

Mar 23, 2025 · Your 'Lua Script : Cheat Table' got so much lines of codes ? this is not a problem anymore ! The better way is to load '.lua' files directly inside the Cheat Table, there is an exemple :

Cheat Engine :: View topic - Speedhack

Apr 8, 2024 · Cheat Engine :: View topic - Speedhack

Cheat Engine :: View topic - DBK error. ALT possible fix?

Apr 18, 2025 · Cheat Engine :: View topic - DBK error. ALT possible fix?

Cheat Engine :: View topic - CE background through Lua

May 9, 2025 · hey guys, im trying to develop a custom theme for my CE through lua files but im having a few issues trying to get everything working. So far ive got most of the main bits accomplished but im having issues with a few small things like replacing the logo and such. if anyone has any suggestions id be immensely grateful as im brand new to lua so its been slow ...

[HELP] I've tried all I know on this game - Cheat Engine

Mar 18, 2025 · I've only just started using Cheat Engine for more than the insanely basic task of finding addresses about a week ago. I've been trying to create a pointer to reuse later in a cheat table for the game. The game is modded, and on a modded server, you can modify points client sidedly, which still ends up saving on the servers side. I discovered this by changing the value ...

Cheat Engine :: View topic - luacode in 7.6 not working

Apr 1, 2025 · The following code works fine in CE 7.5 but in 7.6 it does not print anything. Anyone know how to fix?

Cheat Engine :: View topic - Unable to use DBVM?

Apr 18, 2025 · Back to top Xcuze1337 How do I cheat? Reputation: 0 Joined: 14 Apr 2025 Posts: 5
Posted: Fri Apr 18, 2025 3:51 pm Post subject: Dark Byte wrote: Then i don't know. Maybe check the bios again and read the options. Maybe it says "disable intel-VT" which you set to enabled ?
Virtualization is definitely on according to task manager performance tab ...

Unlock the power of your Linux system with our comprehensive cheat sheet for Linux commands. Master essential commands effortlessly. Learn more now!

[Back to Home](#)