Essential Linux Commands [~100]

Reference Guide

IT & Security Foundation

Table of Contents

Introduction

Linux commands are the foundation of IT and cybersecurity operations. With Linux powering servers, cloud infrastructure, security systems, and IoT devices worldwide, mastering these commands is essential for system administrators, developers, and security analysts.

This reference guide covers approximately 100 essential Linux commands organized into 10 key categories. Each command includes syntax, common options, and practical examples to help you navigate, troubleshoot, and automate Linux systems effectively.

Why Linux Commands Matter:

- Universal presence in enterprise environments
- Critical for system administration and security analysis
- Essential for automation and scripting
- Foundation for troubleshooting and system optimization

1. Navigation Commands

Master the basics of moving around the Linux filesystem.

Command	Syntax & Options	Description	Example
`pwd`	`pwd`	Print working directory	`pwd`
`ls`	`ls [options] [path]`	List directory contents	`ls -la /home`
`cd`	`cd [directory]`	Change directory	`cd /var/log`
`pushd`	`pushd [directory]`	Save current directory and change to new one	`pushd /tmp`
`popd`	`popd`	Return to previously saved directory	`popd`
`dirs`	`dirs [options]`	Display directory stack	`dirs -v`
`tree`	`tree [options] [path]`	Display directory tree structure	`tree -L 2 /etc`
`which`	`which [command]`	Locate command executable	`which python`
`whereis`	`whereis [options] [command]`	Locate binary, source, and manual files	`whereis ls`
`realpath`	`realpath [path]`	Display absolute path	`realpath ../file.txt`

2. File Management

Essential commands for creating, copying, moving, and deleting files and directories.

Command	Syntax & Options	Description	Example
touch	`touch [options] [file]`	Create empty file or update timestamp	`touch newfile.txt`
cp	`cp [options] source dest`	Copy files or directories	`cp -r /home/user /backup`
mv	`mv [options] source dest`	Move or rename files/directories	`mv oldname.txt newname.txt`
rm	`rm [options] [file/dir]`	Remove files and directories	`rm -rf /tmp/folder`
mkdir	`mkdir [options] [directory]`	Create directories	`mkdir -p /path/to/dir`
rmdir	`rmdir [options] [directory]`	Remove empty directories	`rmdir /empty/folder`
ln	`ln [options] target link`	Create links between files	`ln -s /path/file link`
file	`file [options] [file]`	Determine file type	`file /bin/bash`
stat	`stat [options] [file]`	Display file status and information	`stat /etc/passwd`
basename	`basename [path] [suffix]`	Extract filename from path	`basename /path/to/file.txt`

3. File Viewing

Commands for displaying and examining file contents.

Command	Syntax & Options	Description	Example
`cat`	`cat [options] [file]`	Display entire file content	`cat /etc/passwd`
`less`	`less [options] [file]`	View file with pagination	`less /var/log/syslog`
`more`	`more [options] [file]`	View file page by page	`more /etc/hosts`
`head`	`head [options] [file]`	Display first lines of file	`head -20 /var/log/auth.log`
`tail`	`tail [options] [file]`	Display last lines of file	`tail -f /var/log/syslog`
`wc`	`wc [options] [file]`	Count lines, words, characters	`wc -l /etc/passwd`
`nl`	`nl [options] [file]`	Number lines in file	`nl /etc/fstab`
`hexdump`	`hexdump [options] [file]`	Display file in hexadecimal	`hexdump -C /bin/ls`
`strings`	`strings [options] [file]`	Extract printable strings from binary	`strings /bin/bash`
`od`	`od [options] [file]`	Display file in octal/other formats	`od -x file.bin`

4. Access Permissions

Manage file and directory permissions and ownership.

Command	Syntax & Options	Description	Example
chmod	`chmod [options] mode file`	Change file permissions	`chmod 755 script.sh`
chown	`chown [options] owner file`	Change file ownership	`chown user:group file.txt`
chgrp	`chgrp [options] group file`	Change group ownership	`chgrp admin /var/log`
umask	`umask [mask]`	Set default file permissions	`umask 022`
getfacl	`getfacl [options] file`	Display file access control lists	`getfacl /home/user`
setfacl	`setfacl [options] acl file`	Set file access control lists	`setfacl -m u:john:rw file`
lsattr	`lsattr [options] [file]`	List file attributes	`lsattr /etc/passwd`
chattr	`chattr [options] mode file`	Change file attributes	`chattr +i important.txt`
su	`su [options] [user]`	Switch user	`su - root`
sudo	`sudo [options] command`	Execute commands as another user	`sudo apt update`

5. Process Management

Monitor and control running processes and system resources.

Command	Syntax & Options	Description	Example
ps	`ps [options]`	Display running processes	`ps aux`
top	`top [options]`	Display and update running processes	`top -u username`
htop	`htop [options]`	Interactive process viewer	`htop`
kill	`kill [signal] PID`	Terminate process by PID	`kill -9 1234`
killall	`killall [options] name`	Kill processes by name	`killall firefox`
pkill	`pkill [options] pattern`	Kill processes by pattern	`pkill -f "python script"`
pgrep	`pgrep [options] pattern`	Find process IDs by pattern	`pgrep -u root ssh`
nohup	`nohup command &`	Run command immune to hangups	`nohup ./script.sh &`
jobs	`jobs [options]`	Display active jobs	`jobs -l`
bg	`bg [job]`	Put job in background	`bg %1`

6. Network Commands

Network troubleshooting, monitoring, and configuration tools.

Command	Syntax & Options	Description	Example
ping	`ping [options] host`	Test network connectivity	`ping -c 4 google.com`
curl	`curl [options] URL`	Transfer data from/to servers	`curl -I https://example.com`
wget	`wget [options] URL`	Download files from web	`wget -r https://site.com`
ip	`ip [options] object`	Show/manipulate routing, network devices	`ip addr show`
ifconfig	`ifconfig [interface] [options]`	Configure network interface	`ifconfig eth0`
netstat	`netstat [options]`	Display network connections	`netstat -tulpn`
ss	`ss [options]`	Modern netstat replacement	`ss -tulpn`
nmap	`nmap [options] target`	Network discovery and security auditing	`nmap -sS 192.168.1.0/24`
traceroute	`traceroute [options] host`	Trace packet route to destination	`traceroute google.com`
iptables	`iptables [options] rule`	Configure firewall rules	`iptables -L`

7. Disk Usage

Monitor and manage disk space and filesystem operations.

Command	Syntax & Options	Description	Example
`df`	`df [options] [filesystem]`	Display filesystem disk space usage	`df -h`
`du`	`du [options] [path]`	Display directory space usage	`du -sh /var/log`
`mount`	`mount [options] device dir`	Mount filesystem	`mount /dev/sdb1 /mnt`
`umount`	`umount [options] device/dir`	Unmount filesystem	`umount /mnt`
`lsblk`	`lsblk [options]`	List block devices	`lsblk -f`
`fdisk`	`fdisk [options] device`	Manage disk partitions	`fdisk -l /dev/sda`
`fsck`	`fsck [options] device`	Check and repair filesystem	`fsck /dev/sdb1`
`blkid`	`blkid [options] [device]`	Display block device attributes	`blkid /dev/sda1`
`sync`	`sync`	Flush filesystem buffers	`sync`
`lsof`	`lsof [options] [file]`	List open files	`lsof +D /var/log`

8. Search & Filter

Powerful tools for finding, filtering, and processing text and files.

Command	Syntax & Options	Description	Example
grep	`grep [options] pattern [file]`	Search text using patterns	`grep -r "error" /var/log`
find	`find [path] [expression]`	Search for files and directories	`find /home -name "*.txt"`
locate	`locate [options] pattern`	Find files by name (uses database)	`locate passwd`
awk	`awk 'pattern{action}' file`	Pattern scanning and processing	`awk '{print $1}' /etc/passwd`
sed	`sed [options] 'command' file`	Stream editor for filtering/transforming	`sed 's/old/new/g' file.txt`
sort	`sort [options] [file]`	Sort lines of text	`sort -n numbers.txt`
uniq	`uniq [options] [file]`	Report or omit repeated lines	`sort file.txt | uniq -c`
cut	`cut [options] [file]`	Extract sections from lines	`cut -d: -f1 /etc/passwd`
tr	`tr [options] set1 [set2]`	Translate or delete characters	`tr '[:lower:]' '[:upper:]'`
xargs	`xargs [options] [command]`	Execute command from standard input	`find . -name "*.tmp" | xargs rm`

9. Compression

Archive and compress files for storage and transfer.

Command	Syntax & Options	Description	Example
tar	`tar [options] [file] [path]`	Archive files	`tar -czf backup.tar.gz /home`
gzip	`gzip [options] [file]`	Compress files using gzip	`gzip -9 largefile.txt`
gunzip	`gunzip [options] [file]`	Decompress gzip files	`gunzip archive.gz`
zip	`zip [options] archive files`	Create ZIP archives	`zip -r backup.zip /data`
unzip	`unzip [options] archive`	Extract ZIP archives	`unzip -d /tmp archive.zip`
bzip2	`bzip2 [options] [file]`	Compress files using bzip2	`bzip2 -9 document.txt`
bunzip2	`bunzip2 [options] [file]`	Decompress bzip2 files	`bunzip2 archive.bz2`
xz	`xz [options] [file]`	Compress files using xz	`xz -9 database.sql`
7z	`7z [command] [options] archive`	7-Zip archiver	`7z a -t7z backup.7z /data`
zcat	`zcat [file]`	Display compressed file content	`zcat logfile.gz | grep error`

10. Package Management

Install, update, and manage software packages across different distributions.

Command	Syntax & Options	Description	Example
`apt`	`apt [options] command`	Debian/Ubuntu package manager	`apt update && apt upgrade`
`apt-get`	`apt-get [options] command`	Traditional Debian package manager	`apt-get install nginx`
`dpkg`	`dpkg [options] [package]`	Debian package installer	`dpkg -i package.deb`
`yum`	`yum [options] command`	RHEL/CentOS package manager	`yum install httpd`
`dnf`	`dnf [options] command`	Modern Fedora package manager	`dnf search python`
`rpm`	`rpm [options] [package]`	RPM package manager	`rpm -qa | grep kernel`
`snap`	`snap [command] [options]`	Universal package manager	`snap install discord`
`flatpak`	`flatpak [command] [options]`	Application distribution framework	`flatpak install gimp`
`pacman`	`pacman [options] [package]`	Arch Linux package manager	`pacman -Syu`
`zypper`	`zypper [command] [options]`	openSUSE package manager	`zypper refresh`

Conclusion

Mastering these 100+ essential Linux commands provides you with:

- **System Navigation:** Efficiently move through and explore Linux filesystems
- **File Management:** Handle files, directories, and permissions with confidence
- **Process Control:** Monitor and manage system processes effectively
- **Network Operations:** Troubleshoot and configure network connections
- **System Monitoring:** Track resource usage and system performance
- **Security Operations:** Implement proper access controls and system security

Best Practices for Learning:

- Practice commands in a safe environment first
- Always read man pages: `man [command]`
- Use `--help` flag for quick reference
- Combine commands with pipes for powerful workflows
- Create aliases for frequently used command combinations

Remember: These commands are the foundation of Linux system administration and cybersecurity. Regular practice and real-world application will help you become proficient in managing Linux environments efficiently and securely.