

## Linux Commands Cheat Sheet

### Table of Contents

#### File and Directory Management

- Create Directories
- List Files and Directories
- Change Directory
- Copy Files and Directories
- Move/Rename Files
- Remove Files and Directories
- Create Empty Files

#### File Viewing and Editing

- View File Contents
- View Large Files
- Edit Files with Nano
- Edit Files with Vi/Vim

#### Text Editors Command Summary

#### File Permissions and Ownership

- Understanding Permissions
- Change Permissions (Symbolic)
- Change Permissions (Numeric)
- Change Ownership

#### View Permissions

#### Process Management

- View Running Processes
- Process Information
- Control Processes
- Process Priority

#### Background/Foreground Jobs

#### System Information

- OS and Kernel Information
- Hardware Information
- System Specifications

#### Networking Commands

- Basic Connectivity
- Network Configuration
- DNS and Hostname Resolution
- Port and Network Status
- Network Traffic and Monitoring
- Download and Transfer

#### User and Group Management

- User Management
- Group Management
- Sudo Access

#### Package Management

- Debian/Ubuntu (apt)
- RedHat/CentOS (yum/dnf)
- Generic Package Management

#### Disk and Storage Management

- Disk Usage
- Mount and Unmount
- Filesystem Management
- Partition Management

#### Text Processing

- Grep - Search Text
- Sed - Stream Editor
- Awk - Text Processing
- Cut - Extract Columns
- Sort - Sort Text
- Uniq - Remove Duplicates

#### Compression and Archiving

- Tar - Archive Files
- Gzip - Compress Files
- Bzip2 - High Compression
- Zip - Archive and Compress

Searching and Filtering  
Find - Search for Files  
Locate - Fast File Search  
Which and Whereis  
System Administration  
Shutdown and Reboot  
System Logs  
Services Management  
Firewall Management  
SSH and Remote Access  
SSH Connection  
SSH Key Management  
SSH Configuration  
SCP - Secure Copy  
Rsync - Synchronize Files  
Cron Jobs  
Crontab Basics  
Cron Schedule Format  
Cron Examples  
System Crontab  
Tips and Tricks  
Command Line Productivity  
Useful Shortcuts  
Working with Multiple Commands  
Variables and Environment  
Additional Resources

## Linux Commands Cheat Sheet

Last Updated: 2025-12-29

---

### Table of Contents

1. [File and Directory Management](#)
  2. [File Viewing and Editing](#)
  3. [File Permissions and Ownership](#)
  4. [Process Management](#)
  5. [System Information](#)
  6. [Networking Commands](#)
  7. [User and Group Management](#)
  8. [Package Management](#)
  9. [Disk and Storage Management](#)
  10. [Text Processing](#)
  11. [Compression and Archiving](#)
  12. [Searching and Filtering](#)
  13. [System Administration](#)
  14. [SSH and Remote Access](#)
  15. [Cron Jobs](#)
- 

### File and Directory Management

#### Create Directories

```
# Create a single directory
mkdir my_directory

# Create nested directories
mkdir -p /path/to/nested/directory
```

```
# Create multiple directories
mkdir dir1 dir2 dir3
```

## List Files and Directories

```
# Basic listing
ls

# Long format listing with details
ls -l

# List all files including hidden files
ls -a

# List with human-readable file sizes
ls -lh

# Recursive listing of directories
ls -R

# Sort by modification time
ls -lt

# Reverse sort order
ls -lr

# Combined example
ls -lah
```

## Change Directory

```
# Change to home directory
cd ~
cd

# Change to parent directory
cd ..

# Change to previous directory
cd -

# Change to specific path
cd /path/to/directory
```

## Copy Files and Directories

```
# Copy a file
cp source.txt destination.txt

# Copy file with interactive prompt
cp -i source.txt destination.txt

# Copy directory recursively
cp -r source_dir dest_dir

# Copy preserving permissions and timestamps
cp -p source.txt destination.txt

# Verbose copying (show what's being copied)
cp -v source.txt destination.txt

# Combined example
cp -rpv source_dir dest_dir
```

## Move/Rename Files

```
# Move or rename a file
mv old_name.txt new_name.txt

# Move file to another directory
mv file.txt /path/to/directory/

# Move with interactive prompt
mv -i source.txt destination.txt

# Move directory
mv source_dir dest_dir

# Verbose move
mv -v file.txt new_location/
```

## Remove Files and Directories

```
# Remove a file
rm filename.txt

# Remove file with confirmation
rm -i filename.txt

# Remove multiple files
rm file1.txt file2.txt file3.txt

# Remove directory and its contents recursively
rm -r directory_name

# Force remove without confirmation
rm -f filename.txt

# Verbose removal
rm -v filename.txt

# Combined example (force, recursive, verbose)
rm -rfv directory_name
```

## Create Empty Files

```
# Create empty file
touch filename.txt

# Create multiple files
touch file1.txt file2.txt file3.txt

# Update file timestamp
touch existing_file.txt

# Change access and modification times
touch -t 202512291306 filename.txt
```

---

## File Viewing and Editing

### View File Contents

```
# Display entire file
cat filename.txt

# Display with line numbers
```

```
cat -n filename.txt

# Display multiple files
cat file1.txt file2.txt

# Show non-printing characters
cat -A filename.txt

# Suppress repeated blank lines
cat -s filename.txt
```

## View Large Files

```
# View file one page at a time
less filename.txt
# Navigation: Space=next page, b=prev page, q=quit

# View first 10 lines
head -n 10 filename.txt

# View first 20 lines
head -20 filename.txt

# View last 10 lines
tail -n 10 filename.txt

# View last 20 lines
tail -20 filename.txt

# Follow file in real-time (useful for logs)
tail -f /var/log/system.log

# View middle of file
head -n 50 filename.txt | tail -n 20
```

## Edit Files with Nano

```
# Open file in nano editor
nano filename.txt

# Exit: Ctrl+X, then Y for yes, Enter to confirm

# Useful Nano shortcuts:
# Ctrl+O = Save
# Ctrl+X = Exit
# Ctrl+K = Cut line
# Ctrl+U = Paste
# Ctrl+W = Search
```

## Edit Files with Vi/Vim

```
# Open file in vi/vim
vi filename.txt
vim filename.txt

# Vim modes:
# Press i = Insert mode
# Press Esc = Normal mode
# Type :w = Save
# Type :q = Quit
# Type :wq = Save and quit
# Type :q! = Quit without saving

# Open at specific line
```

```
vim +10 filename.txt

# Search in vim
# In normal mode: /search_term, then n for next

# Replace in vim
# In normal mode: :%s/old/new/g (replace all)
```

## Text Editors Command Summary

```
# Using echo to create/append to files
echo "Content" > filename.txt      # Create/overwrite
echo "Content" >> filename.txt    # Append

# Multi-line input
cat > filename.txt << EOF
Line 1
Line 2
Line 3
EOF

# View with syntax highlighting
cat filename.txt
```

---

# File Permissions and Ownership

## Understanding Permissions

```
drwxr-xr-x owner group size date filename
||| | |
||| |   Others: execute
||| |   Others: write
||| |   Others: read
||| |   Group: execute
||| |   Group: write
||| |   Group: read
||| |   Owner: execute
||| |   Owner: write
||| |   Owner: read
```

## Change Permissions (Symbolic)

```
# Add execute permission for owner
chmod u+x filename.txt

# Remove write permission for group
chmod g-w filename.txt

# Add read permission for others
chmod o+r filename.txt

# Add execute for all
chmod a+x filename.txt

# Remove all permissions then add read for owner
chmod u=r filename.txt

# Combined permissions
chmod u+rwx,g+rx,o+x filename.txt

# Apply to directory and contents
chmod -R u+rwx directory/
```

## Change Permissions (Numeric)

```
# r=4, w=2, x=1
# 7=rwx, 6=rw-, 5=r-x, 4=r--, 3=-wx, 2=-w-, 1--x, 0=---

# Owner: rwx, Group: rx, Others: rx (755)
chmod 755 filename.txt

# Owner: rwx, Group: rw, Others: - (760)
chmod 760 filename.txt

# Owner: rwx, Group: -, Others: - (700)
chmod 700 filename.txt

# All can read/write (666)
chmod 666 filename.txt

# Recursive change
chmod -R 755 directory/
```

## Change Ownership

```
# Change owner
chown new_owner filename.txt

# Change owner and group
chown new_owner:new_group filename.txt

# Change only group
chown :new_group filename.txt

# Change owner of directory and contents
chown -R new_owner:new_group directory/

# Change to user and their primary group
sudo chown user:user filename.txt
```

## View Permissions

```
# Long listing shows permissions
ls -l

# Show permissions in numeric format
stat -c "%A %a %n" filename.txt

# Recursive listing with permissions
ls -lR
```

---

## Process Management

### View Running Processes

```
# Display current shell processes
ps

# List all processes
ps aux

# List all processes in forest/tree format
ps auxf

# Show process tree
```

```
pstree  
  
# Show specific process  
ps aux | grep process_name  
  
# Real-time process monitoring  
top  
  
# Enhanced process viewer  
htop
```

## Process Information

```
# Find process by name  
pgrep process_name  
  
# Find process ID  
pidof process_name  
  
# Get process details  
ps -p 1234 -o pid,ppid,cmd  
  
# Show process priority  
ps -o pid,ni,cmd
```

## Control Processes

```
# Run process in background  
command &  
  
# Suspend running process  
Ctrl+Z  
  
# Resume suspended process in background  
bg  
  
# Resume suspended process in foreground  
fg  
  
# List background jobs  
jobs  
  
# Bring job to foreground  
fg %1  
  
# Send SIGTERM signal (graceful termination)  
kill process_id  
  
# Send SIGKILL signal (force termination)  
kill -9 process_id  
  
# Terminate process by name  
killall process_name  
  
# Send specific signal  
kill -SIGTERM process_id
```

## Process Priority

```
# Run process with low priority (nice value 10)  
nice -n 10 command  
  
# Run with different priority  
nice -n -5 command # Higher priority (negative = higher)
```

```
# Change priority of running process
renice -n 5 -p process_id

# Change priority for all processes of user
renice -n 5 -u username
```

## Background/Foreground Jobs

```
# Run command in background
long_running_command &

# Disown background job
disown %1

# Wait for background job
wait process_id

# Run command and disconnect from terminal
nohup long_command &

# Run multiple commands in background
command1 & command2 & command3 &
```

---

## System Information

### OS and Kernel Information

```
# Linux kernel version
uname

# Detailed system information
uname -a

# Kernel name
uname -s

# Kernel release
uname -r

# Operating system
uname -o

# View /etc/os-release for distro info
cat /etc/os-release

# Pretty OS information
lsb_release -a

# System hostname
hostname

# Change hostname (temporary)
sudo hostname new_hostname

# System uptime
uptime

# Boot time
who -b

# Kernel messages
dmesg | tail -20
```

## Hardware Information

```
# CPU information  
cat /proc/cpuinfo  
  
# Number of CPUs  
nproc  
  
# Memory information  
free -h  
  
# Detailed memory info  
cat /proc/meminfo  
  
# GPU information  
lspci | grep -i gpu  
  
# PCI devices  
lspci  
  
# USB devices  
lsusb  
  
# Block devices  
lsblk  
  
# Disk information  
sudo hdparm -I /dev/sda
```

## System Specifications

```
# BIOS/UEFI information  
sudo dmidecode | head -20  
  
# System board information  
sudo dmidecode -t system  
  
# Processor details  
sudo dmidecode -t processor  
  
# Memory slot information  
sudo dmidecode -t memory  
  
# All system information  
sudo dmidecode
```

---

## Networking Commands

### Basic Connectivity

```
# Ping a host  
ping -c 4 example.com  
  
# Ping with specific packet size  
ping -s 56 example.com  
  
# Continuous ping  
ping example.com  
  
# Check if port is reachable  
telnet example.com 80
```

```
# Network connectivity test
curl -I http://example.com
```

## Network Configuration

```
# Display network interfaces
ip address show

# Or use older command
ifconfig

# Show specific interface
ip address show eth0

# Assign IP address (temporary)
sudo ip address add 192.168.1.100/24 dev eth0

# Remove IP address
sudo ip address delete 192.168.1.100/24 dev eth0

# Enable interface
sudo ip link set eth0 up

# Disable interface
sudo ip link set eth0 down

# Show routing table
ip route

# Add static route
sudo ip route add 192.168.2.0/24 via 192.168.1.1

# Remove route
sudo ip route delete 192.168.2.0/24 via 192.168.1.1

# Set default gateway
sudo ip route add default via 192.168.1.1
```

## DNS and Hostname Resolution

```
# Show DNS servers
cat /etc/resolv.conf

# Query DNS
nslookup example.com

# Dig query
dig example.com

# Dig with specific nameserver
dig @8.8.8.8 example.com

# Reverse DNS lookup
dig -x 8.8.8.8

# Show all DNS records
dig example.com ANY

# Quick hostname lookup
host example.com

# Show local hostname
hostname

# Change hostname temporarily
```

```
sudo hostname new-hostname

# Change hostname permanently
sudo nano /etc/hostname
```

## Port and Network Status

```
# List listening ports
ss -tlnp

# Or use older command
netstat -tlnp

# Show all connections
ss -tulpn

# Listening on TCP
ss -tnlp

# Listening on UDP
ss -unlp

# Show specific port
ss -tlnp | grep :80

# Monitor network traffic
netstat -i

# Show network statistics
ss -s

# Real-time network monitoring
watch -n 1 'ss -tulpn'
```

## Network Traffic and Monitoring

```
# Capture packets with tcpdump
sudo tcpdump -i eth0

# Capture packets to file
sudo tcpdump -i eth0 -w capture.pcap

# Filter by protocol
sudo tcpdump -i eth0 tcp

# Filter by port
sudo tcpdump -i eth0 port 80

# Filter by host
sudo tcpdump -i eth0 host 192.168.1.100

# Monitor bandwidth usage
iftop

# Network statistics
vnstat

# Show real-time connections
sudo lsof -i

# Check specific port connections
sudo lsof -i :80
```

## Download and Transfer

```
# Download file
wget http://example.com/file.zip

# Download with specific name
wget -O newname.zip http://example.com/file.zip

# Continue interrupted download
wget -c http://example.com/largefile.zip

# Curl download
curl -O http://example.com/file.zip

# Curl with output file
curl http://example.com/file.zip -o filename.zip

# SCP copy to remote
scp file.txt user@remote:/path/

# SCP copy from remote
scp user@remote:/path/file.txt .

# Recursive SCP
scp -r directory/ user@remote:/path/
```

---

## User and Group Management

### User Management

```
# Add new user
sudo useradd username

# Add user with home directory
sudo useradd -m username

# Add user with specific shell
sudo useradd -m -s /bin/bash username

# Add user with specific UID and GID
sudo useradd -u 1001 -g 1001 username

# Set password for user
sudo passwd username

# Delete user
sudo userdel username

# Delete user and home directory
sudo userdel -r username

# Modify user information
sudo usermod -c "Full Name" username

# Add user to group
sudo usermod -a -G groupname username

# Change user shell
sudo usermod -s /bin/zsh username

# Lock user account
sudo usermod -L username

# Unlock user account
sudo usermod -U username
```

```

# List all users
cat /etc/passwd

# Show current user
whoami

# Show all logged-in users
who

# Switch user
su - username

# Execute command as another user
sudo -u username command

```

## Group Management

```

# Create group
sudo groupadd groupname

# Create group with specific GID
sudo groupadd -g 1001 groupname

# Delete group
sudo groupdel groupname

# Add user to group
sudo gpasswd -a username groupname

# Remove user from group
sudo gpasswd -d username groupname

# List group members
getent group groupname

# Show groups for user
groups username

# Change group membership
sudo usermod -G group1,group2,group3 username

# List all groups
cat /etc/group

# Change file group ownership
chgrp groupname filename

# Change directory group ownership recursively
chgrp -R groupname directory/

```

## Sudo Access

```

# Check sudo access
sudo -l

# Edit sudoers file (safe way)
sudo visudo

# Allow user to run command without password
# In visudo, add: username ALL=(ALL) NOPASSWD: /path/to/command

# Grant sudo to user
sudo usermod -a -G sudo username

```

```
# Grant full sudo access  
# Add to /etc/sudoers: username ALL=(ALL) ALL
```

---

## Package Management

### Debian/Ubuntu (apt)

```
# Update package list  
sudo apt update  
  
# Upgrade all packages  
sudo apt upgrade  
  
# Full upgrade with dependency changes  
sudo apt full-upgrade  
  
# Install package  
sudo apt install package_name  
  
# Install multiple packages  
sudo apt install package1 package2 package3  
  
# Remove package  
sudo apt remove package_name  
  
# Remove package and configuration  
sudo apt purge package_name  
  
# Search for package  
apt search package_name  
  
# Show package information  
apt show package_name  
  
# List installed packages  
apt list --installed  
  
# Check for updates  
apt list --upgradable  
  
# Clean cache  
sudo apt clean  
  
# Remove unused dependencies  
sudo apt autoremove  
  
# Fix broken dependencies  
sudo apt --fix-broken install
```

### RedHat/CentOS (yum/dnf)

```
# Update packages (dnf is newer)  
sudo dnf update  
  
# Or use yum for older systems  
sudo yum update  
  
# Install package  
sudo dnf install package_name  
  
# Remove package  
sudo dnf remove package_name
```

```

# Search for package
dnf search package_name

# Show package info
dnf info package_name

# List installed packages
dnf list installed

# Check for updates
dnf check-update

# Clean cache
sudo dnf clean all

# List available packages
dnf list available

```

## Generic Package Management

```

# Install from source (general steps)
tar -xzf package.tar.gz
cd package/
./configure
make
sudo make install

# Check package dependencies
ldd /path/to/executable

# List shared libraries
ldconfig -p | grep library_name

# View installed package files
dpkg -L package_name # Debian
rpm -ql package_name # RedHat

```

---

## Disk and Storage Management

### Disk Usage

```

# Disk usage of filesystem
df -h

# Disk usage with different units
df -k      # Kilobytes
df -m      # Megabytes
df -BG     # Gigabytes

# Directory usage
du -sh /path/to/directory

# Recursive directory sizes
du -sh /path/to/directory/*

# Show top 10 largest directories
du -sh /path/* | sort -rh | head -10

# Show disk usage by inode
df -i

# Check space used by specific directory
du -sh /home/username

```

```
# Show usage in tree format
tree -h --du /path/
```

## Mount and Unmount

```
# List mounted filesystems
mount

# Mount filesystem
sudo mount /dev/sda1 /mnt/point

# Mount with read-only
sudo mount -r /dev/sda1 /mnt/point

# Unmount filesystem
sudo umount /mnt/point

# Force unmount
sudo umount -f /mnt/point

# Lazy unmount
sudo umount -l /mnt/point

# Mount all filesystems in /etc/fstab
sudo mount -a

# Show mount information
df -T

# List block devices
lsblk

# Detailed partition information
sudo fdisk -l
```

## Filesystem Management

```
# Create ext4 filesystem
sudo mkfs.ext4 /dev/sda1

# Create swap
sudo mkswap /dev/sda2

# Enable swap
sudo swapon /dev/sda2

# Check filesystem
sudo fsck /dev/sda1

# Resize filesystem
sudo resize2fs /dev/sda1

# Label filesystem
sudo e2label /dev/sda1 "my_drive"

# Check disk for bad sectors
sudo badblocks -v /dev/sda1

# Monitor disk I/O
iostat -x 1

# Show open files and disk usage
lsof +D /path/to/directory
```

## Partition Management

```
# List partitions
sudo fdisk -l

# Interactive partition editor
sudo fdisk /dev/sda

# Create partition table
sudo parted /dev/sda

# Show partition info
sudo parted -l

# Backup partition table
sudo fdisk -l /dev/sda > partition_backup.txt

# Restore from backup
sudo fdisk /dev/sda < partition_backup.txt
```

---

## Text Processing

### Grep - Search Text

```
# Basic grep
grep "pattern" filename.txt

# Case-insensitive search
grep -i "pattern" filename.txt

# Invert match (lines NOT containing pattern)
grep -v "pattern" filename.txt

# Count matching lines
grep -c "pattern" filename.txt

# Show line numbers
grep -n "pattern" filename.txt

# Show filename for matches
grep -H "pattern" filename.txt

# Show only matched part
grep -o "pattern" filename.txt

# Recursive search in directory
grep -r "pattern" /path/to/directory/

# Search with regex
grep -E "pattern1|pattern2" filename.txt

# Show context (lines around match)
grep -C 2 "pattern" filename.txt # 2 lines before and after
grep -B 2 "pattern" filename.txt # 2 lines before
grep -A 2 "pattern" filename.txt # 2 lines after

# Multiple pattern search
grep -e "pattern1" -e "pattern2" filename.txt

# Match whole words only
grep -w "word" filename.txt
```

## Sed - Stream Editor

```
# Basic substitution (first occurrence per line)
sed 's/old/new/' filename.txt

# Global substitution (all occurrences)
sed 's/old/new/g' filename.txt

# Case-insensitive substitution
sed 's/old/new/gi' filename.txt

# Save changes to file
sed -i 's/old/new/g' filename.txt

# Backup original and save changes
sed -i.bak 's/old/new/g' filename.txt

# Delete lines matching pattern
sed '/pattern/d' filename.txt

# Delete specific line
sed '5d' filename.txt

# Delete range of lines
sed '1,5d' filename.txt

# Print specific lines
sed -n '5p' filename.txt

# Print range
sed -n '1,5p' filename.txt

# Insert text before line
sed '5i\\New line text' filename.txt

# Append text after line
sed '5a\\New line text' filename.txt

# Replace entire line
sed '5c\\New content' filename.txt
```

## Awk - Text Processing

```
# Print specific column
awk '{print $1}' filename.txt
awk '{print $2, $3}' filename.txt

# Print with delimiter
awk -F':' '{print $1}' /etc/passwd

# Print with custom output separator
awk '{print $1, $2}' OFS="," filename.txt

# Conditional printing
awk '$2 > 100 {print $1}' filename.txt

# Count occurrences
awk '{count++} END {print count}' filename.txt

# Sum column
awk '{sum += $1} END {print sum}' filename.txt

# Pattern matching
awk '/pattern/ {print}' filename.txt
```

```

# Multiple conditions
awk '$1 == "admin" && $2 > 100 {print}' filename.txt

# Change field value
awk '{$2 = "new"; print}' filename.txt

# Print line number and content
awk '{print NR, $0}' filename.txt

```

## Cut - Extract Columns

```

# Extract first column
cut -d' ' -f1 filename.txt

# Extract multiple columns
cut -d':' -f1,3,5 /etc/passwd

# Extract by character range
cut -c1-10 filename.txt

# Extract from character 5 onwards
cut -c5- filename.txt

# Complement selection (exclude specified columns)
cut -d' ' -f2- filename.txt

```

## Sort - Sort Text

```

# Basic sort
sort filename.txt

# Reverse sort
sort -r filename.txt

# Sort numerically
sort -n numbers.txt

# Reverse numeric sort
sort -rn numbers.txt

# Sort by specific column
sort -k2 filename.txt

# Sort by multiple columns
sort -k2,2 -k3,3 filename.txt

# Sort with custom delimiter
sort -t':' -k1 filename.txt

# Remove duplicates while sorting
sort -u filename.txt

# Case-insensitive sort
sort -f filename.txt

# Sort by month
sort -M filename.txt

# Sort by file size
ls -l | sort -k5 -n

```

## Uniq - Remove Duplicates

```
# Remove duplicate consecutive lines
```

```
uniq filename.txt

# Show only duplicate lines
uniq -d filename.txt

# Show only unique lines
uniq -u filename.txt

# Count occurrences
uniq -c filename.txt

# Case-insensitive
uniq -i filename.txt

# Use with sort
sort filename.txt | uniq

# Sort with duplicate count
sort filename.txt | uniq -c | sort -rn
```

---

## Compression and Archiving

### Tar - Archive Files

```
# Create tar archive
tar -cf archive.tar file1 file2 file3

# Create compressed tar.gz
tar -czf archive.tar.gz directory/

# Create tar.bz2
tar -cjf archive.tar.bz2 directory/

# Create tar.xz
tar -cJf archive.tar.xz directory/

# Extract tar
tar -xf archive.tar

# Extract tar.gz
tar -xzf archive.tar.gz

# Extract tar.bz2
tar -xjf archive.tar.bz2

# Extract tar.xz
tar -xJf archive.tar.xz

# Extract to specific directory
tar -xzf archive.tar.gz -C /destination/

# List contents
tar -tzf archive.tar.gz

# Verbose extraction
tar -xzvf archive.tar.gz

# Update archive
tar -uzf archive.tar.gz file.txt

# Remove file from archive
tar --delete -f archive.tar filename
```

## Gzip - Compress Files

```
# Compress file
gzip filename.txt

# Decompress
gunzip filename.txt.gz

# Or use gzip to decompress
gzip -d filename.txt.gz

# Compress with specific compression level (1-9)
gzip -9 filename.txt

# Keep original file
gzip -k filename.txt

# Show compression info
gzip -l filename.txt.gz

# Compress multiple files
gzip file1.txt file2.txt file3.txt

# Stdout compression
gzip -c filename.txt > filename.txt.gz
```

## Bzip2 - High Compression

```
# Compress file
bzip2 filename.txt

# Decompress
bunzip2 filename.txt.bz2

# Or use bzip2 to decompress
bzip2 -d filename.txt.bz2

# Keep original file
bzip2 -k filename.txt

# Compression levels (1-9)
bzip2 -1 filename.txt # Fastest
bzip2 -9 filename.txt # Best compression
```

## Zip - Archive and Compress

```
# Create zip archive
zip archive.zip file1.txt file2.txt

# Create zip of directory
zip -r archive.zip directory/

# Exclude files
zip -r archive.zip directory/ -x "*.log"

# Extract zip
unzip archive.zip

# Extract to specific directory
unzip archive.zip -d /destination/

# List zip contents
unzip -l archive.zip

# Update zip
```

```
zip -u archive.zip newfile.txt  
  
# Delete from zip  
zip -d archive.zip file.txt  
  
# Test zip integrity  
unzip -t archive.zip
```

---

## Searching and Filtering

### Find - Search for Files

```
# Find by name  
find /path -name "filename"  
  
# Find by name (case-insensitive)  
find /path -iname "filename"  
  
# Find by type  
find /path -type f          # Regular files  
find /path -type d          # Directories  
find /path -type l          # Symbolic links  
  
# Find by size  
find /path -size 100k        # Exactly 100KB  
find /path -size +1M         # Larger than 1MB  
find /path -size -1M         # Smaller than 1MB  
  
# Find by modification time  
find /path -mtime 7          # Modified 7 days ago  
find /path -mtime +7         # Modified more than 7 days ago  
find /path -mtime -7         # Modified less than 7 days ago  
  
# Find by access time  
find /path -atime 7  
  
# Find by change time  
find /path -ctime 7  
  
# Find and execute command  
find /path -name "*.log" -exec rm {} \;  
  
# Find with permission  
find /path -perm 644  
  
# Find by user  
find /path -user username  
  
# Find by group  
find /path -group groupname  
  
# Find empty files  
find /path -empty  
  
# Find all and sort by size  
find /path -type f -exec ls -ls {} \; | sort -k5 -rn  
  
# Find with depth limit  
find /path -maxdepth 2 -name "*.txt"  
  
# Find and show only filename  
find /path -name "*.txt" -printf "%f\n"
```

```
# Find modified in last hour
find /path -mmin -60
```

## Locate - Fast File Search

```
# Search for file
locate filename

# Case-insensitive search
locate -i filename

# Count matches
locate -c filename

# Limit results
locate -n 10 filename

# Update database (needed for locate)
sudo updatedb

# Show database statistics
locate -S
```

## Which and Whereis

```
# Find command location
which command_name

# Find all instances
which -a command_name

# Find command in PATH
whereis command_name

# Find binary, source, manual
whereis -b command_name
whereis -s command_name
whereis -m command_name
```

---

# System Administration

## Shutdown and Reboot

```
# Reboot system
sudo reboot

# Shutdown system
sudo shutdown -h now

# Shutdown after delay
sudo shutdown -h +10 # After 10 minutes

# Schedule shutdown at specific time
sudo shutdown -h 14:30

# Shutdown with message
sudo shutdown -h now "System maintenance"

# Cancel scheduled shutdown
sudo shutdown -c

# Power off
```

```
sudo poweroff
```

```
# Halt system  
sudo halt
```

## System Logs

```
# View system log  
sudo tail -f /var/log/syslog

# View all logs  
sudo tail -f /var/log/messages

# View kernel log  
dmesg

# View recent kernel log  
dmesg | tail -20

# Journal entries (systemd)  
journalctl

# Show logs since last boot  
sudo journalctl -b

# Follow journal in real-time  
sudo journalctl -f

# Show specific service logs  
sudo journalctl -u service_name

# Show logs from specific time  
sudo journalctl --since "2025-12-28 10:00:00"

# Show with priority  
sudo journalctl -p err
```

## Services Management

```
# List services  
systemctl list-units --type=service

# Check service status  
systemctl status service_name

# Start service  
sudo systemctl start service_name

# Stop service  
sudo systemctl stop service_name

# Restart service  
sudo systemctl restart service_name

# Reload service configuration  
sudo systemctl reload service_name

# Enable service on boot  
sudo systemctl enable service_name

# Disable service on boot  
sudo systemctl disable service_name

# Show service details  
systemctl show service_name
```

```
# List enabled services
systemctl list-unit-files --type=service

# View service file
systemctl cat service_name
```

## Firewall Management

```
# UFW (Ubuntu Firewall)
sudo ufw status

# Enable firewall
sudo ufw enable

# Disable firewall
sudo ufw disable

# Allow port
sudo ufw allow 22/tcp

# Deny port
sudo ufw deny 22/tcp

# Allow from IP
sudo ufw allow from 192.168.1.100

# Delete rule
sudo ufw delete allow 22/tcp

# Set default policy
sudo ufw default deny incoming
sudo ufw default allow outgoing

# Firewalld (RedHat/CentOS)
sudo firewall-cmd --state

# Add port
sudo firewall-cmd --add-port=80/tcp

# Remove port
sudo firewall-cmd --remove-port=80/tcp

# Reload rules
sudo firewall-cmd --reload
```

---

## SSH and Remote Access

### SSH Connection

```
# Connect to remote server
ssh username@hostname

# Connect with specific port
ssh -p 2222 username@hostname

# Connect with key file
ssh -i /path/to/key.pem username@hostname

# Verbose output
ssh -v username@hostname

# Very verbose
```

```

ssh -vv username@hostname

# Execute remote command
ssh username@hostname "command"

# X11 forwarding
ssh -X username@hostname

# Port forwarding
ssh -L 8080:localhost:80 username@hostname

# Remote port forwarding
ssh -R 8080:localhost:80 username@hostname

# SOCKS proxy
ssh -D 1080 username@hostname

```

## SSH Key Management

```

# Generate SSH key
ssh-keygen -t rsa -b 4096

# Generate with specific name
ssh-keygen -t ed25519 -f ~/.ssh/mykey -C "comment"

# Show public key
cat ~/.ssh/id_rsa.pub

# Copy key to server
ssh-copy-id -i ~/.ssh/id_rsa.pub username@hostname

# Or manually add to authorized_keys
cat ~/.ssh/id_rsa.pub | ssh username@hostname "mkdir -p ~/.ssh &&
cat >> ~/.ssh/authorized_keys"

# Set correct permissions
chmod 700 ~/.ssh
chmod 600 ~/.ssh/id_rsa
chmod 644 ~/.ssh/id_rsa.pub

# Add key to ssh-agent
ssh-add ~/.ssh/id_rsa

# List keys in agent
ssh-add -l

# Remove key from agent
ssh-add -d ~/.ssh/id_rsa

# Remove all keys
ssh-add -D

```

## SSH Configuration

```

# Edit SSH config
nano ~/.ssh/config

# Example config:
# Host myserver
#   HostName example.com
#   User username
#   IdentityFile ~/.ssh/mykey
#   Port 2222

# Check configuration syntax

```

```
ssh -G hostname  
  
# Test configuration  
ssh -T git@github.com
```

## SCP - Secure Copy

```
# Copy file to remote  
scp file.txt username@hostname:/path/  
  
# Copy from remote  
scp username@hostname:/path/file.txt .  
  
# Copy with specific port  
scp -P 2222 file.txt username@hostname:/path/  
  
# Copy directory  
scp -r directory/ username@hostname:/path/  
  
# Copy between remote servers  
scp -3 user1@host1:/file.txt user2@host2:/path/
```

## Rsync - Synchronize Files

```
# Sync to remote  
rsync -avz /local/path/ username@hostname:/remote/path/  
  
# Sync from remote  
rsync -avz username@hostname:/remote/path/ /local/path/  
  
# Exclude files  
rsync -avz --exclude='*.log' /local/path/  
username@hostname:/remote/path/  
  
# Delete files not in source  
rsync -avz --delete /local/path/ username@hostname:/remote/path/  
  
# Dry run  
rsync -avz --dry-run /local/path/ username@hostname:/remote/path/  
  
# Progress and rate limit  
rsync -avz --progress --bwlimit=1000 /local/path/  
username@hostname:/remote/path/  
  
# Show what changed  
rsync -avzi /local/path/ username@hostname:/remote/path/
```

---

## Cron Jobs

### Crontab Basics

```
# Edit crontab  
crontab -e  
  
# View current crontab  
crontab -l  
  
# Remove crontab  
crontab -r  
  
# View user's crontab  
sudo crontab -u username -l
```

```

# Edit user's crontab
sudo crontab -u username -e

# Install crontab from file
crontab /path/to/cronfile

# Save crontab to file
crontab -l > cron_backup.txt

```

## Cron Schedule Format

```

# Minute (0-59)
# | Hour (0-23)
# | | Day of month (1-31)
# | | | Month (1-12)
# | | | | Day of week (0-7, where 0 and 7 are Sunday)
# | | | | |
# * * * * * command_to_execute

# Examples:
0 0 * * *      # Daily at midnight
0 12 * * *     # Daily at noon
0 0 * * 0      # Weekly on Sunday at midnight
0 0 1 * *      # Monthly on 1st at midnight
0 0 1 1 *      # Yearly on Jan 1 at midnight

# Every 5 minutes
*/5 * * * * command

# Every hour
0 * * * * command

# Every 30 minutes
*/30 * * * * command

# Every hour from 9-17
0 9-17 * * * command

# Multiple times
0 9,12,15,18 * * * command

# Every weekday at 2 AM
0 2 * * 1-5 command

```

## Cron Examples

```

# Backup script daily
0 2 * * * /home/user/backup.sh

# Cleanup logs weekly
0 3 * * 0 /usr/local/bin/cleanup_logs.sh

# Check system status hourly
0 * * * * /usr/bin/check_status.sh >> /var/log/check_status.log 2>&1

# Update system weekly
0 4 * * 0 apt update && apt upgrade -y

# Run script with output logging
30 2 * * * /path/to/script.sh >> /var/log/script.log 2>&1

# Run as specific user
0 * * * * root /path/to/command

```

```

# Run every minute
* * * * * /path/to/script.sh

# Run every 2 hours
0 */2 * * * /path/to/script.sh

# Multiple commands
0 0 * * * command1 && command2 && command3

# Conditional execution
0 0 * * * test -f /var/lock/mylock && /path/to/command

```

## System Crontab

```

# System-wide crontab
sudo nano /etc/crontab

# System cron jobs directory
ls /etc/cron.d/
ls /etc/cron.hourly/
ls /etc/cron.daily/
ls /etc/cron.weekly/
ls /etc/cron.monthly/

# Add script to daily cron
sudo cp script.sh /etc/cron.daily/
sudo chmod +x /etc/cron.daily/script.sh

```

---

## Tips and Tricks

### Command Line Productivity

```

# View command history
history

# Search in history
history | grep search_term

# Execute last command
!!

# Execute last command with sudo
sudo !!

# Execute previous command with substitution
^old^new^

# Create alias
alias ll='ls -lah'

# View all aliases
alias

# Remove alias
unalias ll

# Search and execute from history
Ctrl+R (then type search term)

# Clear history
history -c

```

## Useful Shortcuts

```
# Ctrl+C: Interrupt/cancel command
# Ctrl+D: Exit terminal/logout
# Ctrl+L: Clear screen
# Ctrl+A: Move to beginning of line
# Ctrl+E: Move to end of line
# Ctrl+U: Clear from cursor to beginning
# Ctrl+K: Clear from cursor to end
# Tab: Auto-complete
# Ctrl+R: Reverse history search
# Ctrl+Z: Suspend process
# !!: Last command
# !$: Last argument of previous command
# !*: All arguments of previous command
```

## Working with Multiple Commands

```
# Command sequencing
command1; command2; command3

# Conditional execution (AND)
command1 && command2

# Conditional execution (OR)
command1 || command2

# Pipe output
command1 | command2

# Redirect output
command > file.txt          # Overwrite
command >> file.txt         # Append
command 2> error.txt        # Redirect stderr
command &> all.txt          # Redirect stdout and stderr

# Input redirection
command < input.txt

# Command substitution
echo $(date)
echo `date`
```

## Variables and Environment

```
# View all environment variables
env

# Export variable
export VARIABLE_NAME="value"

# Use variable
echo $VARIABLE_NAME

# Permanent environment variable
echo 'export VARIABLE_NAME="value"' >> ~/.bashrc
source ~/.bashrc

# View PATH
echo $PATH

# Add to PATH
export PATH=$PATH:/new/path

# Variable in command
```

```
mkdir $(date +%Y%m%d)
```

---

## Additional Resources

- **Official Manuals:** `man command_name`
  - **Command Help:** `command_name --help`
  - **Online Documentation:** <https://linux.die.net/>
  - **Regular Expression Guide:** `man 7 regex`
  - **Cron Schedule Helper:** <https://crontab.guru/>
- 

**Note:** Most commands that modify system files or settings require `sudo` privileges. Always exercise caution when running commands with administrative access.

---

*This cheat sheet was last updated on 2025-12-29 13:06:59 UTC*