

all unix commands with examples download download

Published: September 19, 2026 | Index ID: #-

All Unix Commands with Examples: A Download Friendly Guide

Unix and Unix like operating systems—Linux, macOS, FreeBSD, and others—are built around a powerful set of command line utilities. These commands let you manage files, monitor processes, configure network settings, and even download data from the internet. Whether you're a beginner learning the ropes or a seasoned sysadmin looking for a quick refresher, this guide presents a comprehensive collection of Unix commands, complete with practical examples and tips on how to download the tools or data you need.

Why Master Unix Commands?

Unlike graphical user interfaces, the command line offers granular control, scripting capabilities, and the ability to automate repetitive tasks. Knowing the right command at the right time can save hours of work. In this article, we'll walk through the most frequently used Unix commands, categorize them by function, and show you how to download the resources they interact with.

Table of Contents

1. File and Directory Operations
2. Process and System Monitoring
3. Text Processing and Manipulation
4. Networking and Remote Access
5. Package Management and Software Installation
6. Security and Permissions
7. Shell Scripting Basics
8. Downloading Files and Software
9. Advanced Topics
10. Conclusion

1. File and Directory Operations

Every Unix user starts with basic file manipulation. Below are the most essential commands, each accompanied by an example that demonstrates how to use it in a real scenario.

1.1 ls – List Directory Contents

ls displays files and directories. Use options to tailor the output.

- ls -l – Long format, showing permissions, owner, size, and modification date.
- ls -a – Include hidden files (those starting with a dot).
- ls -R – Recursively list subdirectories.

Example: ls -l /var/log lists detailed logs in the /var/log directory.

1.2 cd – Change Directory

Navigate the file system.

- cd ~ – Go to your home directory.

- `cd ..` – Move up one level.
- `cd /usr/local/bin` – Jump directly to `/usr/local/bin`.

1.3 `pwd` – Print Working Directory

Shows the current directory path.

Example: `pwd` might output `/home/user/projects`.

1.4 `mkdir` – Make Directories

Create one or more directories.

- `mkdir new_folder` – Create a single folder.
- `mkdir -p parent/child/grandchild` – Create nested directories.

1.5 `rm` – Remove Files or Directories

Delete files or directories. Use with caution.

- `rm file.txt` – Delete a single file.
- `rm -r folder` – Recursively delete a directory and its contents.
- `rm -f file.txt` – Force deletion without confirmation.

1.6 `cp` – Copy Files and Directories

Duplicate files or directories.

- `cp source.txt destination.txt` – Copy a file.
- `cp -r src_dir dest_dir` – Recursively copy a directory.

1.7 `mv` – Move or Rename

Move files or rename them.

- `mv oldname.txt newname.txt` – Rename a file.
- `mv file.txt /tmp/` – Move a file to `/tmp`.

1.8 `touch` – Create Empty Files or Update Timestamps

Create a new empty file or update its access and modification times.

- `touch newfile.txt` – Create an empty file.
- `touch -c file.txt` – Do not create if it doesn't exist.

1.9 `chmod` – Change File Permissions

Set read, write, and execute permissions.

- `chmod 755 script.sh` – Owner read/write/execute, group and others read/execute.
- `chmod u+x file` – Add execute permission for the owner.
- `chmod g-w file` – Remove write permission from the group.

1.10 `chown` – Change File Ownership

Assign a new owner or group to a file or directory.

- `chown user:group file.txt` – Change owner and group.
- `chown -R user:group /path/to/dir` – Recursively change ownership.

1.11 `ln` – Create Hard and Symbolic Links

Create links to files.

- `ln -s /usr/bin/python3 /usr/local/bin/python` – Create a symbolic link.
- `ln /path/to/file /path/to/link` – Create a hard link.

1.12 find – Search for Files

Locate files by name, size, modification time, etc.

- `find /home/user -name "*.txt"` – Find all text files.
- `find . -mtime -7 -type f` – Find files modified in the last week.
- `find / -size +10M` – Find files larger than 10 MB.

1.13 grep – Search Text Patterns

Search within files for specific patterns.

- `grep "error" logfile.log` – Find lines containing “error”.
- `grep -r "TODO" .` – Recursively search for “TODO” in current directory.
- `grep -i "warning"` – Case insensitive search.

1.14 sed – Stream Editor

Perform text transformations on files or input streams.

- `sed 's/old/new/g' file.txt` – Replace all occurrences of “old” with “new”.
- `sed -n '1,10p' file.txt` – Print lines 1 through 10.
- `sed -i 's/foo/bar/g' file.txt` – In place replacement.

1.15 awk – Pattern Scanning and Processing

Process and analyze text files.

- `awk '{print $1}' file.txt` – Print the first field of each line.
- `awk -F:' ' '{print $1, $2}' /etc/passwd` – Print username and password field.
- `awk '/error/ {print $0}' logfile.log` – Print lines containing “error”.

1.16 tar – Archive Files

Compress or decompress file archives.

- `tar -cvzf backup.tar.gz /home/user` – Create a gzipped archive.
- `tar -xvzf backup.tar.gz` – Extract a gzipped archive.
- `tar -tf backup.tar.gz` – List contents of an archive.

1.17 zip / unzip – ZIP Compression

Compress files into ZIP format.

- `zip -r archive.zip folder/` – Recursively zip a folder.
- `unzip archive.zip` – Extract the archive.

1.18 df – Disk Free Space

Display filesystem disk space usage.

- `df -h` – Human readable format.
- `df -T` – Show filesystem type.

1.19 du – Disk Usage

Show disk usage of files and directories.

- `du -sh /var/log` – Summarize size of /var/log.

- `du -h --max-depth=1` – Show sizes of subdirectories.

1.20 head / tail – View File Beginnings and Ends

Inspect the first or last lines of a file.

- `head -n 20 file.txt` – First 20 lines.
- `tail -f logfile.log` – Follow the end of a log file in real time.

2. Process and System Monitoring

Unix provides powerful tools to inspect and control running processes, as well as to monitor system resources.

2.1 ps – Process Status

Show information about active processes.

- `ps aux` – Display all processes for all users.
- `ps -ef` – Alternative format.

Baca Juga (Artikel Terkait)

- [affiliate marketing learn how to make your first 1000 passive income online affiliate marketing for beginners make money online affiliate program internet marketing work from home](http://www.aussizzlegal.com/affiliate-marketing-learn-how-to-make-your-first-1000-passive-income-online-affiliate-marketing-for-beginners-make-money-online-affiliate-program-internet-marketing-work-from-home.pdf)

URL: <http://www.aussizzlegal.com/affiliate-marketing-learn-how-to-make-your-first-1000-passive-income-online-affiliate-marketing-for-beginners-make-money-online-affiliate-program-internet-marketing-work-from-home.pdf>

- [affiliate marketing by harsh](http://www.aussizzlegal.com/affiliate-marketing-by-harsh.pdf)

URL: <http://www.aussizzlegal.com/affiliate-marketing-by-harsh.pdf>

- [affiliate how to make money with amazon affiliate marketing affiliate marketing affiliate marketing 2015 affiliate marketing for beginners amazon program amazon marketing business](http://www.aussizzlegal.com/affiliate-how-to-make-money-with-amazon-affiliate-marketing-affiliate-marketing-affiliate-marketing-2015-affiliate-marketing-for-beginners-amazon-program-amazon-marketing-business.pdf)

URL: <http://www.aussizzlegal.com/affiliate-how-to-make-money-with-amazon-affiliate-marketing-affiliate-marketing-affiliate-marketing-2015-affiliate-marketing-for-beginners-amazon-program-amazon-marketing-business.pdf>

- [affidavit of support under section 213a of the ina form i](http://www.aussizzlegal.com/affidavit-of-support-under-section-213a-of-the-ina-form-i.pdf)

URL: <http://www.aussizzlegal.com/affidavit-of-support-under-section-213a-of-the-ina-form-i.pdf>

Tags: #unix #commands #with #examples #download #download

