

The Linux Command Line Cheat Sheet

The Linux Command Line Cheat Sheet · WebsiteSetup.org

Connect over SSH

The terminal is how you manage a Linux web server directly: files, permissions, logs, and transfers. You reach it by opening a terminal on your machine (macOS Terminal, or WSL / Git Bash on Windows) and connecting over SSH. macOS defaults to the zsh shell, but these Bash commands work the same.

```
ssh user@host
ssh -p 2222 user@host
```

Navigation

Command	What it does	Example
pwd	Print the current directory	pwd
ls -la	List all files, long format	ls -la
ls -lh	Long list, readable sizes	ls -lh
cd	Change directory	cd public_html
cd ..	Move up one level	cd ..
cd ~	Go to home directory	cd ~
cd -	Return to previous directory	cd -
tree	Show folder tree	tree

Files and directories

Command	What it does	Example
touch	Create an empty file	touch index.html
mkdir -p	Make dir with parents	mkdir -p wp/uploads
cp -r	Copy recursively	cp -r site backup
mv	Move or rename	mv old.txt new.txt
rm	Delete a file	rm cache.log
rm -r	Delete a folder	rm -r old_theme
rm -rf	Danger: irreversible, no recycle bin. Force-delete	rm -rf temp
ln -s	Create a symbolic link	ln -s /var/www app
file	Report file type	file logo.png
stat	Show file details	stat wp-config.php

Viewing and editing files

Command	What it does	Example
<code>cat</code>	Print whole file	<code>cat .htaccess</code>
<code>less</code>	Page through a file (q quits)	<code>less error.log</code>
<code>head -n 20</code>	First 20 lines	<code>head -n 20 access.log</code>
<code>tail -f</code>	Follow a live log	<code>tail -f error.log</code>
<code>wc -l</code>	Count lines	<code>wc -l users.csv</code>
<code>nano</code>	Simple editor (Ctrl+X exits)	<code>nano wp-config.php</code>
<code>vim</code>	Editor (:wq save, :q! discard)	<code>vim .htaccess</code>

Search

Command	What it does	Example
<code>grep -r</code>	Search recursively	<code>grep -r "text" .</code>
<code>grep -i</code>	Case-insensitive	<code>grep -i "error" log.txt</code>
<code>grep -rn</code>	Recursive with line numbers	<code>grep -rn "TODO" .</code>
<code>find</code>	Find files by name	<code>find . -name "*.php"</code>
<code>locate</code>	Fast indexed file search	<code>locate nginx.conf</code>
<code>which</code>	Path of a command	<code>which php</code>
<code>whereis</code>	Binary and docs location	<code>whereis php</code>

Permissions and ownership

Permissions are read (4), write (2), execute (1), set per owner, group, and others. So 644 is owner read/write, others read; 755 adds execute for directories.

Command	What it does	Example
<code>chmod 644</code>	Standard file permission	<code>chmod 644 index.php</code>
<code>chmod 755</code>	Standard directory permission	<code>chmod 755 wp-content</code>
<code>chmod +x</code>	Make executable	<code>chmod +x deploy.sh</code>
<code>chown</code>	Change owner and group	<code>chown user:group site</code>
<code>sudo</code>	Run as root	<code>sudo systemctl restart nginx</code>

Danger: chmod 777 is a security hole, not a fix. Use 644 for files and 755 for folders.

Users and system info

Command	What it does	Example
whoami	Your username	whoami
id	User and group IDs	id
who	Who is logged in	who
passwd	Change password	passwd
df -h	Free disk space	df -h
du -sh	Folder size	du -sh wp-content
free -h	RAM usage	free -h
top / htop	Live process view	htop
uname -a	Kernel and system info	uname -a
uptime	Time since boot	uptime

Processes

Command	What it does	Example
ps aux	List all processes	ps aux
top	Live process view	top
kill	Stop a process by ID	kill 4821
kill -9	Force-stop a process	kill -9 4821
killall	Stop by process name	killall php
jobs / bg / fg	Manage shell jobs	fg 1
nohup ... &	Run detached from shell	nohup ./run.sh &
... &	Run in background	./import.sh &

Networking and remote

Command	What it does	Example
ssh	Connect to a server	ssh user@host
scp	Copy file to/from server	scp file user@host:/path
rsync -avz	Sync folders over network	rsync -avz src/ user@host:/dst/
wget	Download from a URL	wget https://ex.com/f.zip
curl -I	Fetch HTTP headers	curl -I https://ex.com
ping	Test if host responds	ping example.com
ss -tulpn	List listening ports	ss -tulpn
dig / nslookup	Look up DNS records	dig example.com

Archives and transfer

Command	What it does	Example
<code>tar -czf</code>	Create a .tar.gz archive	<code>tar -czf a.tar.gz dir</code>
<code>tar -xzf</code>	Extract a .tar.gz archive	<code>tar -xzf a.tar.gz</code>
<code>zip -r</code>	Zip a folder	<code>zip -r site.zip site</code>
<code>unzip</code>	Extract a .zip file	<code>unzip site.zip</code>
<code>gzip</code>	Compress a single file	<code>gzip access.log</code>

Package management

Debian and Ubuntu use apt. RHEL and CentOS use dnf (older systems use yum).

```
sudo apt update
sudo apt install nginx

sudo dnf install nginx
```

Pipes, redirects, and text tools

Command	What it does	Example
<code> </code>	Pipe output to next command	<code>ls wc -l</code>
<code>></code>	Redirect, overwrite file	<code>ls > files.txt</code>
<code>>></code>	Redirect, append to file	<code>echo hi >> log.txt</code>
<code><</code>	Feed file as input	<code>sort < names.txt</code>
<code>&&</code>	Run next if first succeeds	<code>cd site && ls</code>
<code> </code>	Run next if first fails	<code>ping -c1 h echo down</code>
<code>xargs</code>	Build command from input	<code>ls xargs rm</code>
<code>sort</code>	Sort lines	<code>sort names.txt</code>
<code>uniq</code>	Remove adjacent dupes	<code>sort log uniq</code>
<code>cut</code>	Extract columns	<code>cut -d, -f1 data.csv</code>
<code>sed</code>	Find and replace text	<code>sed 's/a/b/g' file</code>
<code>awk</code>	Process columns	<code>awk '{print \$1}' log</code>
<code>echo</code>	Print text or variable	<code>echo \$PATH</code>
<code>history / clear</code>	Show history / clear screen	<code>history</code>

Shortcuts and handy tricks

Command	What it does	Example
<code>man</code>	Open a command's manual	<code>man grep</code>
<code>--help</code>	Quick usage summary	<code>tar --help</code>
<code>Tab</code>	Autocomplete names	<code>cd pub</code> then <code>Tab</code>
<code>Ctrl+C</code>	Cancel running command	<code>Ctrl+C</code>
<code>Ctrl+R</code>	Search history	<code>Ctrl+R</code>
<code>Ctrl+L</code>	Clear screen	<code>Ctrl+L</code>
<code>!!</code>	Repeat last command	<code>!!</code>
<code>sudo !!</code>	Repeat last as root	<code>sudo !!</code>

Site-owner recipes

Watch the live error log (Apache: `/var/log/apache2/error.log`):

```
tail -f /var/log/nginx/error.log
```

Find what is filling the disk:

```
du -ah . | sort -rh | head -20
```

Fix WordPress permissions:

```
find . -type d -exec chmod 755 {} \;  
find . -type f -exec chmod 644 {} \;
```

Upload a file to your server:

```
scp backup.zip user@host:/var/www/html/
```

Search your code:

```
grep -rn "DB_NAME" wp-config.php
```

Always `ls` a folder before you `rm` anything in it.

Dangerous commands and gotchas

- **`rm -rf` is irreversible.** Never run `rm -rf /` or `rm -rf ~`.
- `sudo` runs as root. Double-check before you press Enter.
- Linux paths are case-sensitive and use forward slashes.
- **`chmod 777` is a security risk.** Use 644 and 755 instead.
- macOS uses `zsh`; Windows needs WSL or Git Bash. Bash commands still work.
- Quote paths that contain spaces. Use Tab completion and the up arrow to avoid typos.