

# CheatSheet: Linux Files

# LINUX

- PDF Link: [cheatsheet-file-A4.pdf](#), Category: linux
- Blog URL: <https://cheatsheet.dennyzhang.com/cheatsheet-file-A4>
- Related posts: CheatSheet: Linux Process, CheatSheet: Linux Networking, [#denny-cheatsheets](#)

File me Issues or star this repo.

## 1.1 Check file

Name	Comment
Show file content	<code>cat /etc/hosts</code>
Show file content with line numbers	<code>cat -n /etc/hosts</code>
Show with line numbers, excluding blank lines	<code>cat -b /etc/hosts</code>
Show the first 3 lines	<code>head -n3 /etc/hosts</code>
Show the first 20 bytes	<code>head -c20 /etc/hosts</code>
Show the last 3 lines	<code>tail -n3 /etc/hosts</code>
Show the last 20 bytes	<code>tail -c20 /etc/hosts</code>
Keep tailing log file	<code>tail -f /var/log/system.log, taif /var/log/system.log</code>
Show file starting from 4th line	<code>more +4 /etc/hosts</code>
Show the 4th line	<code>sed -n '4p' /etc/hosts</code>
Show 4th, 5th, 7th and 8th lines	<code>sed -n '4,5p;7,8p' /etc/hosts</code>
Show matched string with 3 lines before and after	<code>grep -C 3 "127.0.0.1" /etc/hosts</code>
For table-like file, show 2nd column	<code>awk -F'\t' '{print \$2}' /etc/hosts</code>
For table-like file, swap 1st and 2nd columns	<code>awk -F'\t' '{print \$2,\$1}' /etc/hosts</code>
Find file encoding	<code>file -i /var/log/corecaptured.log</code>

## 1.2 Find file

Name	Summary
Basic find	<code>find /home/mac/&lt;myfolder&gt; -name "my*.log"</code>
Find files with two patterns	<code>find . -iname "my*.log" -o -iname "my*.txt"</code>
Find folder old than 3 days	<code>find . -maxdepth 1 -type d -ctime +3</code>
Find files changed within 60 minutes	<code>find /var/log -mmin 60 -type f</code>
Find with ls details	<code>find . \( -iname README.md \) -ls</code>
Find files filtered by size	<code>find /var/log -type f -size +50k -size -100k</code>
Find files older than another file	<code>find . -newer /tmp/file</code>
Find files while excluding patterns	<code>find . -name "*.log" -prune -o -name ".git" -prune -o -type f -print0</code>

## 1.3 Find and delete

Name	Summary
Find and delete with given names	<code>find . -name Thumbs.db -delete</code>
Recursively delete empty folders	<code>find . -type d -empty -delete</code>
Delete files haven't been updated in 5 days	<code>find . -mtime +5 -exec rm {} \;</code>
Delete folders created older than 5 days	<code>find . -name "npm-*" -type d -ctime +2 -exec rm -rf {} +</code>

## 1.4 Watch file

Name	Comment
Show file changes	<code>watch -d -n 1 stat /var/log/message</code>
Keep tailing log files	<code>tail -f /var/log/system.log, taif /var/log/system.log</code>

## 1.5 Copy file

Name	Comment
Copy one file	<code>cp /etc/hosts /tmp/hosts</code>
Copy one folder	<code>cp -r /usr/local/bin/ /tmp/bin/</code>
Copy for backup	<code>cp /tmp/hosts{,.bak}, ls -lth /tmp/hosts*</code>
Create a copy but ask confirmation for overwrite	<code>cp -i ~/foo.txt /tmp/foo.txt</code>
Create a copy for backup with timestamp as suffix	<code>cp myfile.txt{,."\$(date +%Y%m%d-%H%M%S)"}&gt;</code>
Copy files by checking timestamp	<code>rsync -av \$src_dir \$dest_dir</code>
Copy files by comparing checksum	<code>rsync -avc \$src_dir \$dest_dir</code>

## 1.6 Watch file

Name	Command
Remove a file or folder	<code>rm -rf &lt;path&gt;</code>
Remove a file or an empty folder	<code>rm -f &lt;path&gt;</code>
Remove a file by inode	<code>find &lt;path&gt; -inum 5555 -exec rm -i '{}' \;</code>

## 1.7 Diff File

Name	Command
Diff two files	<code>diff &lt;file1&gt; &lt;file2&gt;</code>
Ignore uppercase and lowercase	<code>diff -i &lt;file1&gt; &lt;file2&gt;</code>
Diff output of two commands	<code>diff &lt;(date) &lt;(somecommand)</code>
Generate patch from two files	<code>diff -Naur &lt;file1&gt; &lt;file2&gt; &gt; diff.patch</code>
Diff two directories	<code>diff -r &lt;dir1&gt; &lt;dir2&gt;</code>
Show only brief summary	<code>diff -r --brief &lt;dir1&gt; &lt;dir2&gt;</code>

## 1.8 Make directory

Name	Comment
Make directory. Report error, if existing	<code>mkdir foo</code>
Make directory. Avoid reporting error, if existing	<code>mkdir -p foo</code>
Make directory and its parents	<code>mkdir -p foo/bar/dir1</code>
Make directories with the hierarchy	<code>mkdir -p foo/{bar,bad/{dir1,dir2}}, tree foo</code>

## 1.9 More Resources

License: Code is licendiff under MIT License.