SIDEBAR Linux

logrotate Cheat Sheet

Sidebar

Linux · logrotate Cheat Sheet

Keeping log files from filling the disk. Options and directives were checked against the upstream logrotate manual page; logrotate itself was not run here (no Linux machine to hand), so check logrotate --version and man logrotate on yours, and use -d first. The paths shown are the usual Debian/Ubuntu and RHEL ones.

THE BIG PICTURE

`/etc/logrotate.conf`	Main config: global defaults plus an include of logrotate.d
`/etc/logrotate.d/`	One small file per application (nginx, apache2, your own)
`/var/lib/logrotate.status`	State file: when each log was last rotated
`logrotate.timer`	Runs it daily on systemd distros (older ones use cron.daily)
`systemctl list-timers logrotate.timer`	See when it last ran and runs next
`it is not a daemon`	It runs, rotates what is due, and exits

COMMAND LINE

`logrotate -d file`	Debug: dry run, changes nothing, prints what it would do
`logrotate -f file`	Force rotation now, even if not due
`logrotate -v file`	Verbose: say what it is doing
`logrotate -s state file`	Use a different state file (handy for testing)
`logrotate -l log file`	Write the verbose output to a file
`logrotate --version`	Check which version you have
`sudo logrotate /etc/logrotate.conf`	Run everything by hand, as the timer would

A TYPICAL CONFIG

`/var/log/myapp/*.log {`	Which logs (globs allowed), then a block in braces
`daily`	How often
`rotate 14`	Keep 14 old logs, delete older
`compress`	Gzip the old logs
`delaycompress`	Leave the newest old log uncompressed
`missingok`	No error if the log is not there
`notifempty`	Do not rotate an empty log
`create 0640 www-data adm`	New empty log with this mode, owner, group
`}`	End of block

WHEN TO ROTATE

`hourly`	Needs logrotate itself to be run hourly (cron or timer)
`daily weekly monthly yearly`	Time-based rotation
`size 100M`	Rotate when over this size (k, M, G): ignores the time rules
`maxsize 100M`	Rotate when over this size even before the time is up
`minsize 1M`	Rotate only if over this size AND the time is up
`minage 3`	Skip logs younger than 3 days
`maxage 30`	Delete rotated logs older than 30 days (checked when rotating)

KEEPING & NAMING

`rotate N`	Old logs kept; 0 deletes them, -1 keeps them (use maxage)
`dateext`	app.log-20270317, not app.log.1
`dateformat -%Y-%m-%d`	Own date format (%Y %m %d %H %M %S and a few more)
`extension .log`	Keep the extension: app.1.log
`olddir /var/log/old`	Keep old logs in another folder (same disk)
`createolddir 0755 root root`	Make that folder if missing
`start 0`	First number to use (app.log.0)

COMPRESSION

`compress / nocompress`	Compress old logs or not
`delaycompress`	Wait one cycle; needs compress. Use if the app keeps writing to the old file
`compresscmd /usr/bin/xz`	Different compressor
`compressext .xz`	Matching extension
`compressoptions -9`	Options for the compressor

THE FILE IS STILL OPEN

`create`	Default way: rename the log, make a new one, tell the app to reopen
`postrotate … endscript`	Commands to run after rotating (e.g. reload the service)
`copytruncate`	Copy the log, then empty the original: for apps that cannot reopen
`sharedscripts`	Run the script once for all logs matched, not once each
`prerotate / firstaction`	Run before one log / before any log
`lastaction / preremove`	Run after all logs / before a log is deleted

SAFETY & OWNERSHIP

`su www-data adm`	Rotate as this user and group, not root
`missingok`	Do not fail when the log is absent
`ifempty / notifempty`	Rotate empty logs / skip them
`include /etc/logrotate.d`	Read more config; must not be group/world-writable
`allowhardlink`	Rotate files with more than one hard link (off by default)
`shred`	Overwrite deleted logs with shred -u

RECIPES

`logrotate -d /etc/logrotate.d/myapp`	Test a new config safely before trusting it
`logrotate -fv /etc/logrotate.d/myapp`	Force one rotation and watch it happen
`grep myapp /var/lib/logrotate.status`	When did it last rotate this log?
`ls -l /var/log/myapp/`	Check the result: .1, .2.gz and so on
`size 50M + rotate 5 + compress`	Cap a busy log at about 50M plus 5 old ones
`systemctl status logrotate`	Did the last scheduled run succeed?

THE #1 TRAP: TEST WITH -D, AND MIND THE OPEN FILE

Run `logrotate -d` before you trust a new config: it shows what would happen and changes nothing. And remember that renaming a log does not make the program stop writing to it: without a `postrotate` reload or `copytruncate`, the app keeps writing into the renamed file and the new one stays empty. Also, a log directory writable by a non-root user makes logrotate complain about "insecure permissions": add a `su user group` line.