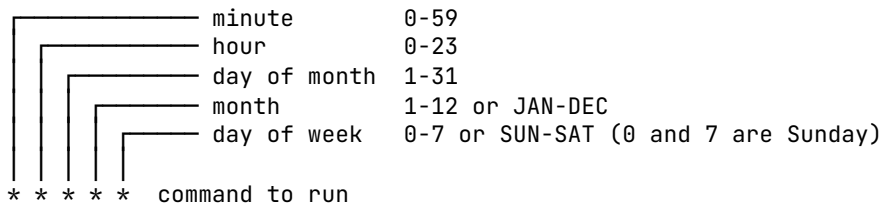
$ cd reference / Cron syntax

Cron syntax and crontab reference

This page describes the crontab format of Vixie cron and cronie, which most Linux distributions ship. Other schedulers borrow the syntax and change details.

[01] The five fields



A user crontab (`crontab -e`) has these five fields and the command. A system crontab in `/etc/crontab` or `/etc/cron.d/` has a sixth field, the user name, between the schedule and the command.

[02] Values, ranges, steps, lists

SYNTAX	MEANING	EXAMPLE
`*`	every value	`* * * * *` every minute
`5`	one value	`5 * * * *` at minute 5 of every hour
`1-5`	inclusive range	`0 9 * * 1-5` 09:00 on weekdays
`1,15`	list	`0 0 1,15 * *` midnight on the 1st and 15th
`*/15`	every 15th value from the start of the range	`*/15 * * * *` at :00, :15, :30, :45
`9-17/2`	step within a range	`0 9-17/2 * * *` 09, 11, 13, 15, 17
`MON-FRI`	names, any case, in month and weekday fields	`0 8 * JAN,JUL MON`

[03] Shortcuts

SHORTCUT	EQUIVALENT
`@yearly` / `@annually`	`0 0 1 1 *`
`@monthly`	`0 0 1 * *`
`@weekly`	`0 0 * * 0`
`@daily` / `@midnight`	`0 0 * * *`

SHORTCUT	EQUIVALENT
`@hourly`	`0 * * * *`
`@reboot`	once, when the cron daemon starts

[04] The day rule

If both the day-of-month and the day-of-week field are restricted (neither starts with `*`), a day matches when *either* field matches. If one of them is `*`, only the other one counts.

EXPRESSION	RUNS
`0 0 13 * 5`	every 13th of the month and every Friday
`0 0 13 * *`	every 13th
`0 0 * * 5`	every Friday

To run only on Friday the 13th, schedule `0 0 13 * *` and test the weekday in the command: `[ "$(date +\%u)" = 5 ] && job`.

[05] Steps do not cross boundaries

`*/7` in the day-of-month field gives the 1st, 8th, 15th, 22nd and 29th, then starts again on the 1st. The gap between the 29th and the next 1st is two or three days, not seven. `*/40` in the minute field means minutes 0 and 40, so the job runs 20 minutes apart across the hour. For a true fixed interval, use a systemd timer with `OnUnitActiveSec=` or check elapsed time inside the job.

[06] Why a job does not run

CAUSE	FIX
A bare `%` in the command is turned into a newline	escape it: `date +\%F`
PATH is minimal, often `/usr/bin:/bin`	use full paths or set `PATH=` at the top of the crontab
Commands run with `/bin/sh` and no login profile: no aliases, nothing from .bashrc or .profile	call a script that sets up what it needs, or set `SHELL=` and variables at the top of the crontab
The server clock is in a different zone than you assumed, often UTC	check `timedatectl`; cronie supports `CRON_TZ=`
On Debian and Ubuntu, a file in /etc/cron.d is skipped if its name contains a dot	use only letters, digits, `-` and `_` in the name; keep it owned by root and not writable by group or others
Output went to local mail nobody reads	redirect: `job >> /var/log/job.log 2>&1`

CAUSE	FIX
The last line has no trailing newline	end the crontab with a newline

[07] Overlapping runs

cron starts a job on schedule whether or not the previous run has finished. Wrap long jobs in a lock so that a slow run is skipped rather than doubled:

```
*/5 * * * *  flock -n /run/lock/report.lock /usr/local/bin/report
```

Daylight saving changes cause the other kind of surprise. Vixie cron and cronie handle shifts of less than three hours specially: a fixed-time job in the skipped hour runs right after the clocks move forward, and no job is run twice when they move back. Wildcard schedules such as hourly jobs simply follow the clock, and other schedulers may skip or repeat a run. Schedule in UTC, or avoid the hour between 01:00 and 03:00.

[08] Managing crontabs

TASK	COMMAND
Edit your crontab	`crontab -e`
List it	`crontab -l`
Edit another user's crontab (root)	`crontab -u deploy -e`
See whether jobs ran (systemd)	`journalctl -u cron -u crond --since today`
Back up before removing	`crontab -l > crontab.bak && crontab -r`

[09] Sources

[1] crontab(5), cronie

[2] cron(8), cronie: clock changes

[3] cronie source, entry.c: field parsing and the * rule for the day fields

[4] cronie source, cron.c: what runs when the clock jumps

[5] cron(8), Debian: file names in /etc/cron.d