# How to Set Up a Cron Job to Run a Script Automatically

The simplest automation tool that already exists on every Linux server

## What a Cron Job Is

Cron is a scheduler built into Linux. You tell it: run this command at this time, on this schedule. It does it, every time, without you touching anything. Every scraper, backup, report, and automated email in your infrastructure probably runs via cron.

## The Crontab Syntax

The format looks strange the first time. It's five time fields followed by the command:

```
# ■■ minute (0-59)
# ■ ■■ hour (0-23)
# ■ ■ ■■ day of month (1-31)
# ■ ■ ■ ■■ month (1-12)
# ■ ■ ■ ■ ■■ day of week (0=Sun, 6=Sat)
# ■ ■ ■ ■ ■
```

> • ○ ▪ ▪ ▪ command to run

## Common Schedule Examples

| Schedule | Cron Expression |
|---|---|
| Every minute | * |
| Every hour at :00 | 0**** |
| Every day at 8 AM | 08*** |
| Every day at midnight | 00*** |
| Every Monday at 9 AM | 09**1 |

Every 15 minutes */15 * * * *

Educational only. Not legal advice. No attorney-client relationship. ©2026 Chariton Media.

> First day of every month at 6 AM 061**

## How to Edit Your Crontab

```
crontab -e # open your crontab for editing
crontab -l # list your current cron jobs
# Example entry  -  run a Python script every day at 8 AM:
```

> 0 8 * * * /usr/bin/python3 /home/user/myscript.py >> /home/user/logs/myscript.log 2>&1

## The >> log 2>&1 Part

This is how you capture output. Without it, cron runs your script silently and you have no idea what happened. >> appends stdout to the log file. 2>&1 also sends stderr (errors) there. Always include this so you have something to read when things break.

## Three Common Gotchas

> Gotcha 1; Use full paths. Cron doesn't know your PATH. Use /usr/bin/python3 not

```
python3. Use /home/user/script.py not ~/script.py.
```

> Gotcha 2; Cron uses server time. If your server is UTC and you're Central, 8 AM Central = 2 PM UTC. Do the math before setting the schedule. Gotcha 3; Test the command manually first. Run the exact command in a terminal before putting it in cron. If it doesn't work there, it won't work in cron either. Using AI to Help With This You don't have to fully understand this to use it. Here are prompts that work: "Write a cron job that runs /home/user/scraper.py every day at 7 AM and logs output to /home/user/logs/scraper.log."

"My cron job is not running. Here is my crontab entry: [paste]. What is wrong with it?"

## 6:30 PM."

"Convert this schedule to a cron expression: every weekday at 6:30 AM and

Educational only. Not legal advice. No attorney-client relationship. ©2026 Chariton Media.