

# Running with cron (scheduled commands)

› Sometimes you want the bot to do some stuff in the background: maintenance, error reports, database cleanup...

Let's say we have a command that sends logs (if they exist) and a command that cleans up the database from records older than the number given as a parameter.

NOTE: This page assumes usage of Linux's **cron**.

› Let's begin:

- Copy example `cron.php` to a place of your choosing
- Make sure vendor path is correct
- Set basic configuration - just like in other hook/getUpdates scripts ( `API KEY`, `BOT USERNAME`, `MYSQL connection details...` )
- Set any other options ( `Download/Upload paths`, `command configs` ) here as well (for `API KEY`, `BOT USERNAME`, `MYSQL connection details...` )
- Set the commands you want to run - `$commands = ['/sendlogs', '/cleanup 30'];`
- Add cron entry - `*/* * * * * cd /home/user/bot/; /usr/local/bin/php -q cron.php > logs/cron.log 2>&1` (will print out output to `logs/cron.log`, you can change this to path of your choosing)

› Now every minute the script will run commands:

- `' sendlogs '` without any parameters
- `' cleanup '` with parameter `' 30 '`

--

› Advices:

› Having the command act as admin and cron command at the same time:

In commands made to be run by admin and a cron if you want the script to show output in the console (for `cron.log` ) you should consider having something like this at the bottom of the command (assumes `$data['text']` hold the output):

```
if ($message->getFrom()->getUsername() != $this->getTelegram()->getBotName()) {
    return Request::sendMessage($data);
} else {
    echo $data['text'] . PHP_EOL;
    return Request::emptyResponse();
}
```

This will:

- allow the command to be run by user/admin and return output as a message (for example - running on demand database cleanup by admin and show number of affected rows)
- show command output in the console if run from cron (which will be written to `cron.log`) and don't send the message to the bot itself (which will return an error from Telegram API, bot cannot send message to itself)

› **Check if command is run from the cron**

- When using Webhook you can check if your command is run by cron by using `defined('STDIN')` check, this constant is only defined when script is run from CLI (will return `false` on the webspace - requests from Telegram API)!

› **How to get the 'parameter' inside command?**

Just like for other commands - `$text = trim($message->getText(true));`

› **Sending text message instead of running a command**

- `$commands = ['your text message that triggers something', '/cleanup 30];`