

About Cron jobs

Cron Jobs in cPanel allow you to automate recurring tasks by scheduling commands or scripts to run at specific times. This is especially useful for automating maintenance processes such as clearing temporary files, running database backups, or sending scheduled reports.

By using Cron Jobs, you can ensure important scripts run consistently and on time—without manual effort.

1. Accessing the Cron Jobs Tool

To begin setting up automated tasks:

1. Log in to your cPanel account.
2. Scroll to the **Advanced** section.
3. Click **Cron Jobs**.



This opens the Cron Jobs interface where you can create new scheduled tasks or manage existing ones.

2. Understanding How Cron Jobs Work

A **cron job** is a time-based task that runs a command or script automatically, according to a schedule you define. You control when a task runs by setting time intervals using five fields:

- **Minute** (0-59)
- **Hour** (0-23)
- **Day** (1-31)
- **Month** (1-12)
- **Weekday** (0-7, where 0 and 7 represent Sunday)

For example:

To run a task at **2:00 AM every day**, use the following settings:

<code>makefile</code>	
-----------------------	--

minute	0
Hour	2
Day	*
Month	*
Weekday	*

Common Cron Job Schedules

Schedule	Settings
Once Daily	Minute: 0, Hour: [your hour], Day: *, Month: *, Weekday: *
Every Hour	Minute: 0, Hour: *, Day: *, Month: *, Weekday: *
Weekly	Minute: 0, Hour: 3, Day: *, Month: *, Weekday: 1 (Monday)

3. Adding a New Cron Job

To schedule a new task:

Step 1: Configure Email Notifications

- If you'd like to receive an email every time the cron job runs, enter your email address in the **Cron Email** field.
- If you **do not** want to receive emails for a particular job, add this to the end of the command:

javascript:

```
>/dev/null 2>&1
```

This prevents your inbox from being flooded with cron output.

Cron Jobs

Cron jobs allow you to automate certain commands or scripts on your site. You can set a command or script to run at a specific time every day, week, etc. For example, you could set a cron job to delete temporary files every week to free up disk space.

Warning: You need to have a good knowledge of Linux commands before you can use cron jobs effectively. Check your script with your hosting administrator before adding a cron job.

Cron Email

Send an email with the output every time the cron job runs. [less >](#)

You can have cron send an email every time it runs a command which produces output. If you do not want an email to be sent for an individual cron job, you can redirect the command's output to /dev/null. For example: `mycommand >/dev/null 2>&1`

Current Email
newsite

Email

[Update Email](#)

Step 2: Set the Time Intervals

- Use the dropdown menus to select when you want your command to run.
- You can customize this based on minute, hour, day, month, and weekday.

Add New Cron Job

PHP command examples:

General example:
`/usr/local/bin/php /home/newsite/public_html/path/to/cron/script`

Domain-specific example:
`/usr/local/bin/ea-php99 /home/newsite/domain_path/path/to/cron/script`

In the above example, replace "ea-php99" with the PHP version assigned to the domain you wish to use. Look in the [MultiPHP Manager](#) for the actual PHP version assigned to a domain.

Common Settings

-- Common Settings --

Minute:
 -- Common Settings --

Hour:
 -- Common Settings --

Day:
 -- Common Settings --

Month:
 -- Common Settings --

Weekday:
 -- Common Settings --

Command:

Step 3: Enter the Command

In the **Command** field, enter the script you want to execute. Examples:

General PHP Command:

swift

```
/usr/local/bin/php /home/username/public_html/path/to/cron/script.php
```

PHP Command with Custom Version (Domain-Specific):

swift

```
/usr/local/bin/ea-php81 /home/username/public_html/path/to/cron/script.php
```

Replace `ea-php81` with the PHP version assigned to your domain (check via **MultiPHP Manager** in cPanel).

Step 4: Add the Cron Job

Click **Add New Cron Job**. The task will now appear in your list of current cron jobs.

4. Managing Existing Cron Jobs

Once created, your cron jobs will appear in the **Current Cron Jobs** section.

Command:

[Add New Cron Job](#)

Current Cron Jobs

Minute	Hour	Day	Month	Weekday	Command	Actions
No Cron Jobs						

This table displays the following information:

Minute	Hour	Day	Month	Weekday	Command	Actions
*	2	*	*	*	/usr/local/bin/php /home/newsite/public_html/path/to/cron/script	[Edit] [Delete]

Each entry shows:

- Scheduled time (minute, hour, day, month, weekday)
- The command to be run
- Action options: **Edit** or **Delete**

You can use these options to:

- Modify the schedule or command of an existing job
- Remove jobs that are no longer needed

5. Cron Job Email Notifications

By default, cPanel will send an email with the output of your cron job to the email listed in the **Current Email** field. This is useful for monitoring whether tasks complete successfully.

- To change the notification email, update the address in the **Cron Email** section at the top of the Cron Jobs interface.
- To disable email notifications for a specific job, append the command with:

javascript:

```
>/dev/null 2>&1
```

6. Best Practices for Using Cron Jobs

- **Test Scripts First:** Always run your script manually before scheduling it to ensure it works as expected.
- **Use Secure Commands:** Make sure the script or command you enter doesn't contain errors or unsafe operations.

- **Avoid Unnecessary Jobs:** Only schedule tasks that are needed. Overusing cron jobs can affect server performance.
- **Monitor Output:** Check your email reports (or logs) to ensure tasks run successfully.
- **Use Full Paths:** Always specify the full path to your script or command to avoid errors.
- **Know Your Hosting Limits:** Some hosting providers limit how often cron jobs can run or how many are allowed. Contact your provider if you're unsure.

Cron Jobs in cPanel are a powerful way to automate routine or recurring website tasks. Whether you're managing backups, sending email reports, or maintaining databases, cron jobs can help you save time and ensure tasks are performed consistently.

Take time to carefully configure your job schedules, use full command paths, and monitor performance to get the most from this feature.

Revision #1

Created 28 March 2025 06:30:22 by Admin

Updated 28 March 2025 06:42:51 by Admin