Scheduled Scans, the Email Digest and Critical Alerts

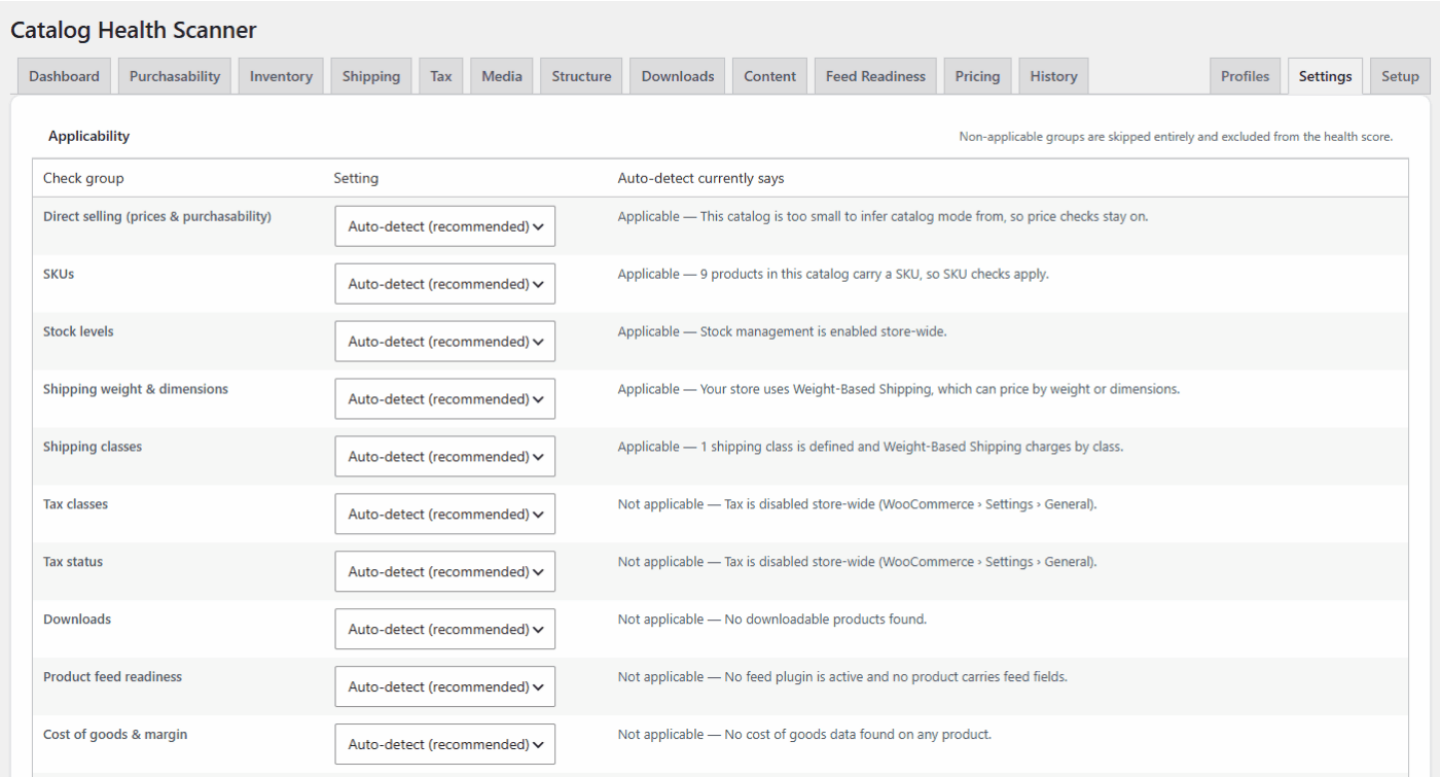
How to · https://wpfactory.com/docs/scheduled-scans-email-digest/

Scheduled scans and the email digest

Scheduled scans let Catalog Health Scanner for WooCommerce Pro audit your catalog on its own and email you only when something has actually changed. Catalog Health Scanner sends nothing when the catalog is clean, so an inbox with no message from it is a meaningful signal rather than a gap you have to check.

Where to set it up

Go to **WPFactory > Catalog Health > Settings** and find the **Scheduled scans & email digest** panel. Everything on this page is saved into a single option, `wpfchs_schedule`.



Setting	Key in `wpfchs_schedule`	Type	Default
Run scans automatically	`enabled`	yes / no	no
Frequency	`frequency`	daily / weekly / monthly	weekly
Profile	`profile`	Any built-in or custom profile id	`revenue_blockers`
Email digest	`digest_enabled`	yes / no	yes
Digest recipients	`digest_recipients`	Comma-separated emails	Your site admin email
Immediate critical alerts	`alerts_enabled`	yes / no	no
Alert recipients	`alert_recipients`	Comma-separated emails	Your site admin email
Auto-fix after scheduled scans	`auto_fix_enabled`	yes / no	no

Setting the schedule

1. Tick **Run scans automatically**.
2. Choose a **Frequency**: Daily, Weekly, or Monthly.
3. Choose the **Profile** the scheduled scan should use. Every built-in and custom profile is in the list.
4. Click **Save settings**.

Saving reschedules the cron event immediately: the existing `wpfchs_scheduled_scan_event` is cleared and, if scheduling is enabled, a new one is registered for one hour from now on the interval you picked. Turning scheduling off clears the event entirely.

One schedule is active at a time. It has one frequency and one profile, which is the right shape for the job: a trend line is only comparable when every scan asks the same question, and mixing profiles in one history makes the score movement meaningless. Pick the profile you want to be measured on, usually **Revenue Blockers** for routine monitoring or **Full Scan** for a periodic deep audit, and run

other profiles manually when you need them.

Notes on frequency

Daily and weekly map to the WordPress core cron intervals of the same name. **Monthly** is not a core WordPress interval: if nothing on your site registers a `monthly` schedule, the plugin falls back to weekly rather than silently never running. If you need a true monthly cadence, register a `monthly` interval through the standard `cron_schedules` filter and it will be used.

Scheduled scans use WP-Cron, which is triggered by site traffic unless your host runs a real cron job. On a very quiet site, a scheduled scan can start later than the nominal time. If timing matters, switch to a server-side cron in the usual WordPress way.

How a scheduled scan runs

The cron event starts the scan under the configured profile with the trigger recorded as `scheduled`, then hands the work to the scanner's own stepping events (`wpfchs_scan_step_event`), so a large catalog is processed in batches rather than in one long request. If a scan is already running when the event fires, the scheduled run is skipped rather than stacked on top.

When the scan completes, `wpfchs_scan_completed` fires and three things happen in order: optional auto-fix, the critical alert, then the digest. The order is deliberate, so the email you receive reflects the catalog after any automatic fixes rather than before them.

The email digest

It is only sent when there is something new

The digest is not a "your scan finished" notification. It is sent only when the completed scheduled scan found issues that were seen for the first time in that scan. If nothing new turned up, nothing is sent. Silence means a clean catalog, which is what makes the digest worth reading rather than worth filtering.

The digest is also limited to scheduled scans. Running a scan by hand from the dashboard never triggers it, because you are already looking at the result.

The subject line leads with the number

Subjects are written so the important part survives a notification preview on a phone. They are prefixed with your site name, then:

Situation	Subject
New critical issues were found	"7 products cannot be sold correctly"
New issues, none critical	"23 new catalog issues found"

The number is the concrete count from that scan, never a vague "issues were found".

What is in the body

- The health score as a heading, with the direction of travel against the previous completed scan ("up from 78%" or "down from 91%").
- The new issues grouped by category, and within each category by check, with the number of affected products.
- A link on every check name that opens that check's filtered view in the admin.
- A link to the Catalog Health dashboard.

The per-check links are real deep links: they carry the category tab and the check id, so clicking one lands you on the exact group of products rather than on a dashboard you then have to navigate from. That is the difference between an email you act on and an email you archive.

Recipients

Both recipient fields accept several addresses separated by commas. Whitespace and semicolons are tolerated on input, and each address is validated and normalised on save, so an invalid entry is dropped rather than breaking the send. If a field ends up empty, no email is sent for that channel.

Immediate critical alerts

The digest tells you what changed since the last scan. Critical alerts exist for the one case where waiting for a digest is too slow: a product that has just become unsellable.

Tick **Email as soon as a scan finds new critical issues** and set the alert recipients. This is a separate channel with its own recipient list, so a developer or agency can receive criticals without being subscribed to the routine digest, or the other way round.

Unlike the digest, critical alerts fire after **any** completed scan, including one you started by hand. Anything that produces a new critical finding produces the alert.

The alert:

- Uses the subject "Critical: N products just became unsellable", with N being the concrete count.
- Lists up to twenty affected products, each linked to its product edit screen, with the name of the check that failed.
- Ends with a link to the Catalog Health dashboard, where the critical banner tells you which category the criticals are in.

It only reports criticals seen for the first time in that scan. A critical that has been open for weeks does not re-alert every time a scan runs, so the alert stays a signal instead of becoming a recurring nag.

Optional auto-fix after scheduled scans

The last setting in the panel opts you in to unattended fixing: **Automatically apply the unambiguous "quick win" fixes after each scheduled scan**. It is off by default.

When on, a completed scheduled scan applies exactly the auto tier of fixes: expired sale prices, dead gallery and linked-product references, stock-status sync, and the rest of the one-correct-answer repairs. Bulk-assignable fixes, which need a value from you, are never run automatically.

The safety model is unchanged. Every change is written as its own batch in the fix log with the before and after values, and stays undoable from the History tab for the length of the undo window (30 days by default). It runs before the digest is built, so the email reflects the post-fix state, and the per-product counters are refreshed so the dashboard agrees with it. Full detail in fix issues safely.

Review the fix log after the first unattended run. It is reversible, but you should know what it does to your data before you leave it running.

Cron events and hooks

Name	Type	Purpose
`wpfchs_scheduled_scan_event`	Recurring cron event	Starts the scheduled scan. Registered and cleared automatically when `wpfchs_schedule` is saved.
`wpfchs_scan_step_event`	Single cron event	Processes the next batch of an unattended scan, rescheduled until the scan finishes.
`wpfchs_scan_completed`	Action	Fires with the scan id after any scan completes. Auto-fix, alerts and digest are all hooked to it.
`wpfchs_auto_remediated`	Action	Fires after unattended auto-fix actually changed something, with the scan id and the result.

Troubleshooting

Symptom	Likely cause
No digest arrived	Most often there was nothing new to report, which is the intended behaviour. Check the History tab: if a scan completed and your issue counts did not grow, no email was due.
No scan ran at all	WP-Cron needs traffic. Check the History tab for completed scans, and consider a server-side cron on a low-traffic site.
Monthly behaves like weekly	No `monthly` cron interval is registered on the site, so the plugin falls back rather than never running.
Emails are not delivered	The plugin sends through `wp_mail()`. If other WordPress emails also fail, the problem is your site's mail configuration, not the scanner.
A scheduled scan was skipped	Another scan was still running when the event fired. The next scheduled run proceeds normally.

Related reading

- Scan profiles
- Scan history and comparison
- Settings reference

Need help?

If scheduled scans run but the digest never arrives even when new issues appear, contact WPFactory support with your schedule settings and the last few rows of the History tab.



CATALOG HEALTH SCANNER FOR WOOCOMMERCE

See every feature and the pricing on the Catalog Health Scanner for WooCommerce plugin page.

View plugin →

Related articles

- Run Your First Catalog Scan
- The Catalog Health Dashboard and Score
- Ignore False Positives and Manage the Ignore List
- Check Applicability: Why Some Checks Are Skipped
- Triage Catalog Issues in the Category Tabs