

# Cron Configuration

## Proxmox KVM module **WHMCS**

[Order now](#) | [Download](#) | [FAQ](#)

## Overview

The module requires a cron system to process VM deployments, package changes, backups, and other automated tasks. Two cron modes are available, and you can choose the one that best fits your environment.

## Cron Modes

### Mode 1: WHMCS Hook (Default)

In this mode, the module hooks into the standard WHMCS cron and executes its tasks automatically each time the WHMCS cron runs.

#### **Advantages:**

- No additional configuration required
- Works out of the box after module activation
- Uses the existing WHMCS cron schedule

**When to use:** This is the recommended mode for most installations. If your WHMCS cron runs every 5 minutes (the standard recommendation), this provides timely task execution.

No additional crontab entries are needed. Just ensure the standard WHMCS cron is running:

```
*/5 * * * * php -q /path/to/whmcs/cron/cron.php
```

# Mode 2: Standalone

In this mode, the module's cron runs independently from the WHMCS cron via a separate crontab entry. This gives you independent control over the module's cron frequency.

## Advantages:

- Independent schedule from WHMCS cron
- Can run more frequently for faster VM provisioning
- Useful if your WHMCS cron runs less frequently

**When to use:** Use standalone mode if you need the module to process tasks more frequently than your WHMCS cron runs, or if you want to separate the module's workload from the main WHMCS cron.

To set up standalone cron, add the following to your server's crontab:

```
* /5 * * * * php -q /path/to/whmcs/modules/addons/puq_proxmox_kvm/cron.php
```

PUQ Proxmox KVM

Home IP Pools DNS Zones VM Management Settings Help v3.0

### Cron Settings

**Cron Mode**

WHMCS Hook — Cron tasks run automatically via WHMCS AfterCronJob hook (~every 5 min)

Standalone Cron File — Add cron.php to system crontab (runs every minute, intervals managed internally)

**Crontab Setup**

Add the following line to your system crontab ( `crontab -e` ):

```
* * * * * php /home/whmcs-dev-puq-p1/web/whmcs-dev-puq-p1/public_html/modules/addons/puq_proxmox_kvm/cron.php > /dev/null 2>&1
```

CLI usage: `php cron.php --help` for all options. Run single task: `php cron.php --task-processVirtualMachines` Show status: `php cron.php --list`

**Task Intervals**

Set execution interval for each cron task in minutes. Set to 0 to disable a task. Intervals apply to both WHMCS Hook and Standalone modes.

Task	Interval (min)	Last Run	Status
<b>Process Virtual Machines</b> Continues VM provisioning (deploy, clone, set IP, CPU/RAM, disk, network, cloudinit). Also handles package changes and DNS record updates for VMs not yet in "ready" state.	1 default: 1	2026-04-10 03:34:00	ACTIVE
<b>Remove Old Snapshots</b> Removes snapshots that have exceeded their configured lifetime. Only processes VMs with existing snapshots.	1 default: 5	2026-04-10 03:34:00	ACTIVE
<b>Restore Backup Status</b> Monitors progress of backup restore operations. Sends confirmation email when restore completes successfully.	1 default: 1	2026-04-10 03:34:00	ACTIVE
<b>Now Backup Status</b> Monitors progress of on-demand backup operations. Clears tracking when backup completes or fails.	1 default: 1	2026-04-10 03:34:00	ACTIVE
<b>Schedule Backup</b> Triggers scheduled backups based on per-VM daytime configuration. Automatically rotates old backups when limit is reached.	1 default: 5	2026-04-10 03:34:00	ACTIVE
<b>Collecting Statistics</b> Collects network traffic statistics (in/out) from Proxmox RRD data. Runs once per hour per VM. Auto-cleans records older than 35 days.	60 default: 60	2026-04-10 03:34:00	ACTIVE

**Concurrency Protection**

Lock Timeout (seconds):

If a cron process runs longer than this, the lock is considered stale and will be automatically removed on the next run. This prevents stuck locks from blocking all future cron executions. Default: 600 (10 min).

# Configuring Cron Mode

The cron mode is configured in the addon settings:

1. Navigate to **Addons > PUQ Proxmox KVM**

2. Go to the **Settings** page
3. Select the **Cron** tab
4. Choose your preferred cron mode: **WHMCS Hook** or **Standalone**
5. Save settings

PUQ Proxmox KVM

**Cron Settings**

**Cron Mode**

- WHMCS Hook** — Cron tasks run automatically via WHMCS AfterCronJob hook (~every 5 min)
- Standalone Cron File** — Add cron.php to system crontab (runs every minute, intervals managed internally)

**Task Intervals**

Set execution interval for each cron task in minutes. Set to 0 to disable a task. Intervals apply to both WHMCS Hook and Standalone modes.

Task	Interval (min)	Last Run	Status
<b>Process Virtual Machines</b> Continues VM provisioning (deploy, clone, set IP, CPU/RAM, disk, network, cloudinit). Also handles package changes and DNS record updates for VMs not yet in "ready" state.	1 default: 1	2026-04-10 03:34:00	ACTIVE
<b>Remove Old Snapshots</b> Removes snapshots that have exceeded their configured lifetime. Only processes VMs with existing snapshots.	1 default: 5	2026-04-10 03:34:00	ACTIVE
<b>Restore Backup Status</b> Monitors progress of backup restore operations. Sends confirmation email when restore completes successfully.	1 default: 1	2026-04-10 03:34:00	ACTIVE
<b>Now Backup Status</b> Monitors progress of on-demand backup operations. Clears tracking when backup completes or fails.	1 default: 1	2026-04-10 03:34:00	ACTIVE
<b>Schedule Backup</b> Triggers scheduled backups based on per-VM day/time configuration. Automatically rotates old backups when limit is reached.	1 default: 5	2026-04-10 03:34:00	ACTIVE
<b>Collecting Statistics</b> Collects network traffic statistics (in/out) from Proxmox RRD data. Runs once per hour per VM. Auto-cleans records older than 35 days.	60 default: 60	2026-04-10 03:34:00	ACTIVE

**Concurrency Protection**

**Lock Timeout (seconds)**

600

If a cron process runs longer than this, the lock is considered stale and will be automatically removed on the next run. This prevents stuck locks from blocking all future cron executions. Default: 600 (10 min).

# Task Intervals

Each cron task has a configurable interval that controls how often it runs. These intervals can be adjusted in the Cron settings page. For details on individual tasks and their intervals, see the [Cron and Automation](#) section.

# Verifying Cron Operation

To confirm the cron is running correctly:

1. Navigate to the addon **Settings > Cron** page
2. Check the **Last Run** timestamp for each task
3. Verify there are no stale lock files

If tasks are not executing, check:

- The WHMCS cron is running (for Hook mode)
  - The standalone crontab entry is correct (for Standalone mode)
  - PHP CLI is available at the path specified in the crontab
  - File permissions allow the cron script to execute
- 

Revision #7

Created 10 April 2026 19:07:37 by Ruslan

Updated 15 May 2026 14:07:11 by Ruslan