

Release Notes Docs

- [2.0.0 \(Major\)](#)
- [1.4.1](#)
- [1.4.0 \(Major\)](#)
- [1.3.7](#)
- [1.3.6](#)
- [1.3.5](#)
- [1.3.4](#)
- [1.3.3](#)
- [1.3.2](#)
- [1.3.0 \(Major\)](#)
- [1.2.8 to 1.3.0 - Upgrade](#)

2.0.0 (Major)

Architectural changes

Go

- Backend rewritten from the ground up in Go, replacing Node.js and Next.js. The stack is more scalable, uses less RAM, and is significantly more performant.
- [sqlc](#) is used for type-safe SQL against PostgreSQL (compile-time checked queries instead of ad-hoc ORM access patterns).
- [golang-migrate](#) is used for database migrations (replacing Prisma ORM).
- Structured logging with the standard library `log/slog` for cleaner, machine-parseable logs in production.

Background jobs and automation

- Background work is handled by [asynq](#) on Redis instead of BullMQ. PatchMon no longer ships the embedded **Bull Board** stack; queue visibility and triggers live in the existing **Automation** UI, which reduces attack surface, image size, and operational complexity.

Docker

- Docker is the officially supported deployment method going forward; bare-metal installs are discontinued. A migration document describes the upgrade path.
- Hardened base images are used. They ship with near-zero CVEs and a smaller footprint.
- No separate frontend container: static React build artifacts are embedded in the Go binary. The container runs that binary (by default on port 3000) with `chi: /api/*` is handled by the server, so nginx inside PatchMon is no longer required. You still use nginx or another reverse proxy in front for TLS termination and public access as usual.
- A Guacamole (guacd) sidecar is included for Windows RDP. It is separate for now; RDP/VNC for Windows is an area we intend to improve.

API documentation

- **OpenAPI 3** spec is served at `/api/v1/openapi.json`, with **Swagger UI** under `/api/v1/api-docs` (authenticated) for exploring integration endpoints.

New features

- **Linux patching:** Deploy updates per host or in bulk, on demand or on a schedule. **Policies** support host/group assignments and exclusions; runs support **approval, stop, retry validation**, and **live log streaming** over WebSocket.
- **Microsoft Windows agent** (beta) and **FreeBSD** agent support.
- **Windows Updates** (beta Windows agent): server APIs for update results, reboot state, superseded cleanup, and approved-guid sync, aligned with the new Windows agent.
- **Advanced monitoring & alerting:** richer alert lifecycle (including assignment and bulk actions), optional **advanced alert configuration** for tuning and cleanup where your edition includes it.
- **Notifications:** first-class **destinations** (SMTP, webhooks, ntfy), **routes, delivery log**, and **scheduled reports** so operational signals leave PatchMon reliably.
- **Environment variables in the GUI:** many settings that were previously only in process environment can be **viewed and edited from the Settings UI** (per-key updates, with sensible validation), so you change less by hand in compose or shell env for day-to-day tuning.
- **OIDC / SSO:** configure OpenID Connect from the same Settings area, including **import from environment** when you are migrating from a file-based or container env setup.

Other improvements

- **Compliance / OpenSCAP:** SSG and CIS benchmarking content is **bundled in the server binary** at build time. Agents no longer pull scanning content from GitHub; everyone shares one versioned source of truth and less outbound traffic from agents.
- **SSO:** improved sign-in flows and **Entra ID** integration compared to 1.4.x OIDC edge cases (e.g. redirect loops with auto SSO).
- **Dashboard:** additional cards and data surfaces; dashboard layout preferences carry forward in the new UI.
- **Host integration config: apply pending config** from the server so integration changes are applied to agents in a controlled, observable way.
- **Settings reliability:** server URL and related configuration are reimplemented on the Go stack with database-backed resolution, addressing classes of “settings did not persist” issues from the Node era.
- **Reverse proxy awareness:** continued correct use of forwarded headers for HTTPS/WSS behind proxies (without the Bull Board-specific HTTP quirks from 1.4.x).
- **Optional admin pprof:** when enabled, CPU/memory profiling endpoints are available to administrators for performance investigation.

Packaging and editions

- Features are grouped into **capability modules** (e.g. patching policies, advanced alerts, custom branding, Docker inventory, compliance depth, AI assist, remote access). Core workflows stay simple; larger deployments can enable more surface area where their **subscription or license** allows. See in-app **Context** / billing documentation for your tenant.

Known issues

- **Remote Desktop (RDP)**: there is a known bug with the RDP connection flow in this release. A fix is planned for the next release.

Migrations

This covers migration for Docker, Proxmox community scripts, and legacy `setup.sh` installs:

[Migrating from 1.4.2 to 2.0.0](#)

1.4.1

? PatchMon 1.4.1

A maintenance release with OIDC improvements, FreeBSD agent support, installer fixes, and various bug fixes and improvements.

? OIDC Improvements and Hot Fixes

- OIDC authentication fixes and stability improvements
- Hot fixes for edge cases in SSO flows

?? FreeBSD Agent Support

- **Native FreeBSD agent support** — run the PatchMon agent on FreeBSD hosts
- Initial FreeBSD support via community contribution

? Native Installer Upgrade Fixes

- Fixes for native installer upgrade paths
- Improved reliability when upgrading existing installations

? Host Table Views Not Saving -> Bug Fix

- Fixed an issue where host table view preferences (columns, sort order, filters) were not being saved
- Table view state now persists correctly across sessions

? Agent Memory Leaks and Improvements

- Addressed memory leaks in the agent
- General agent stability and resource usage improvements

? Better API Integration Scoping

- Improved scoping for Integration API credentials and access
- Tighter integration between API keys and their permitted scope

? Acknowledgements

- **@RuTHlessBEat200** — for agent and OIDC fixes
 - **@mminkus** — for FreeBSD initial PR
 - The rest of the community for their support and help on Discord and GitHub
-

1.4.0 (Major)

? PatchMon 1.4.0

A major release with security compliance scanning, OIDC SSO, an alerting engine, web SSH terminal, and AI-assisted terminal support.

?? Security Compliance Scanning

- **OpenSCAP CIS Benchmark scanning** directly from the agent (Level 1 / Level 2)
- **Docker Bench for Security** when Docker integration is enabled
- **Compliance dashboard** with fleet-wide scores, pass/fail breakdowns, and scan history
- **Optional auto-remediation** of failed rules during scans

? OIDC Single Sign-On

- **OpenID Connect authentication** with Authentik, Keycloak, Okta, or any OIDC provider
- **Automatic user provisioning** on first OIDC login
- **Group-based role mapping** from your identity provider to PatchMon roles
- **Option to disable local auth** and enforce SSO-only login

? Alerting & Reporting

- **New Reporting page** with filtering by severity, type, status, and assignment
- **Host Down alerts** real time view of host uptime
- **Alert types** including server update, agent update, and host down
- **Per-alert-type configuration** for default severity, auto-assignment, escalation, and retention

? Web SSH Terminal

- **Browser-based SSH** to any host from the PatchMon UI
- **Direct and proxy modes** (proxy mode routes through the agent, no SSH port exposure needed)

? AI Terminal Assistant

- **AI chat panel** inside the SSH terminal for command suggestions and troubleshooting
- **Multiple providers** supported: OpenRouter, Anthropic, OpenAI, Google Gemini
- **Context-aware** using your recent terminal output

?? UI Improvements

- **Toast notifications** replacing disruptive `alert()` popups
- **Error boundary** with crash recovery and a copyable error report
- **"Waiting for Connection" screen** with real-time status when onboarding a new host
- **Swagger / OpenAPI docs** served at `/api-docs` on the server

? Other

- **Superuser management permission** (`can_manage_superuser`) for finer-grained RBAC
- **More stats** and details on hosts with added flags such as `?include=stats` or `?updates_only=true`

Plus Much Much More

1.3.7

? ALERT : Auto-update of Agent issue

Versions <1.3.6 have an issue where the service does not restart after auto-update. OpenRC systems are unaffected and work correctly. This means you will unfortunately have to use `systemctl start patchmon-agent` on your systems to load up 1.3.7 agent when it auto-updates shortly.

Very sorry for this, future versions are fixed - I built this release notes notification feature specifically to notify you of this.

? New Features & Improvements :

Mobile UI: Mobile user interface improvements are mostly complete, providing a better experience on mobile devices.

Systemctl Helper Script: In future versions (1.3.7+), a systemctl helper script will be available to assist with auto-update service restarts.

Staggered Agent Intervals: Agents now report at staggered times to prevent overwhelming the PatchMon server. If the agent report interval is set to 60 minutes, different hosts will report at different times. This is in the `config.yml` as "report_offset: xxxx" in seconds

Reboot Detection Information: Reboot detection information is now stored in the database. When the "Reboot Required" flag is displayed, hovering over it will show the specific reason why a reboot is needed (Reboot feature still needs work and it will be much better in 1.3.8)

JSON Report Output: The `patchmon-agent report --json` command now outputs the complete report payload to the console in JSON format instead of sending it to the PatchMon server. This is very useful for integrating PatchMon agent data with other tools and for diagnostic purposes.

Persistent Docker Toggle: Docker integration toggle state is now persisted in the database, eliminating in-memory configuration issues. No more losing Docker settings on container restarts (thanks to the community for initiating this feature).

Config.yml Synchronization: The agent now writes and compares the `config.yml` file with the server configuration upon startup, ensuring better synchronization of settings between the agent and server.

Network Information Page: Enhanced network information page to display IPv6 addresses and support multiple network interfaces, providing more comprehensive network details.

Auto-Update Logic Fix: Fixed an issue where agents would auto-update even when per-host auto-update was disabled. The logic now properly honours both server-wide auto-update settings and per-host auto-update settings.

Prisma Version Fix: Fixed Prisma version issues affecting Kubernetes deployments by statically setting the Prisma version in package.json files.

Hiding Github Version: Added a toggle in Server Version settings to disable showing the github release notes on the login screen

Thank you to all contributors :D

1.3.6

Fixed ProxMox Auto-enrollment script

1.3.5

- Fixed critical Bug relating to auto-update failing and looping in a reboot due to incorrect version checking mechanism.

Especially on x86 or ARM processors, the version checking method was flawed so it kept trying to reinstall the agent.

This release will be further elaborated on but for now marking as latest.

1.3.4

?Fixes and Enhancements

Alpine Support

Version 1.3.4 brings about better apk support for Alpine OS

Auto-enrollment API

In Integration settings you can now create a single command (like a master command) which does not require that you add the host first. This is useful for embedding inside ansible deployment scripts or other use-cases where you have quite a few hosts to add.

image

NOTE: Proxmox api endpoint has changed:

It now goes like this:

```
curl -s "https://patchmon-url/api/v1/auto-enrollment/script?type=proxmox-lxc&token_key=KEY&token_secret=SECRET" | bash
```

Notice that at the end of the auto-enrollment we have a new endpoint called `script` , which then specifies the script type such as proxmox-lxc

Uninstallation command updated and script to remove the instance totally (with the ability to optionally remove backups of agents etc)

Reboot Needed flag

The server now gives a tag and notification if a host needs rebooting due to the kernel version mismatching when installed kernel differs from the running kernel. There is also a new dashboard card that shows this qty in the hosts table.

Other improvements

- Now uses POSIX compatible installation scripts
- Does not use `/bin/bash` , now we use `/bin/sh`
- Added robots.txt to discourage search engines from discovering a public facing version of PatchMon

Upgrading note / instructions

Some members are reporting a upgrade Loop on their systems, please stop the patchmon-agent and start it again

```
systemctl stop patchmon-agent && systemctl start patchmon-agent
```

 after the upgrade. The issue is that the built-in restart function after downloading the binary isn't loading the new binary files so it's using what's loading in cache/memory.

In the newer versions we have introduced a helper-script

Upgrading

Docker

Pull the latest image and bring it up, nothing new needs doing to env or container settings.

Bare metal

```
curl -fsSL -o setup.sh  
https://raw.githubusercontent.com/PatchMon/PatchMon/refs/heads/main/setup.sh && chmod +x setup.sh  
&& bash setup.sh --update
```

Proxmox community Script

Go into the LXC and type in `update` <https://community-scripts.github.io/ProxmoxVE/scripts?id=patchmon>

Agents

Agents will auto-upgrade to 1.3.4 if the settings have been selected to allow this. Pinned release for the agent repo : <https://github.com/PatchMon/PatchMon-agent/releases/tag/1.3.4>

Many thanks to the community for their hard work and support. <3

https://buymeacoffee.com/iby__

1.3.3

?Fixes and Enhancements

ARM support

Supports the installation of ARM and ARM64 agents. Drop down added when creating the command for the installation of the agent and also modified the logic of version handling when the PatchMon server is hosted on an ARM based server. This is because previously the server was checking the current version of its binary but it was pinned to checking the amd64 version of the binary, now this is dynamic based on the actual architecture of the PatchMon server.

Disabling / Enabling docker integration

In the individual hosts page there is now an integrations tab which allows the user to enable or disable docker integration. This amends the `/etc/patchmon/config.yml` with the relevant settings.

image

Dashboard Chart fix

Previously the data taken for this chart was taken from the hosts details data but this did not honor unique packages so the quantities was inflated. Now we have a separate database table model that collects information every 30 minutes for data metrics. This is much more efficient and the charts are now displaying accurate trends.

RHEL fixes

RHEL derived Operating systems such as AlmaLinux, Oracle Linux etc had a bug in the agent that was using the dnf package manager where the version data was not populated in the json payload causing errors upon sending the report. This has now been fixed and also security package quantities are also showing.

TimeZone support

The server environment file now supports a `TIMEZONE=` variable to show things in the right timezone on the app.

Backend container crashing

This was due to error handling not in place when there was docker events that were closed unexpectedly. This has been fixed to handle it correctly.

Ui fixes

Left justification on tables in the repos page Sorting by Security in repos page now fixed

Upgrading

Docker

Pull the latest image and bring it up, nothing new needs doing to env or container settings.

Bare metal

```
curl -fsSL -o setup.sh  
https://raw.githubusercontent.com/PatchMon/PatchMon/refs/heads/main/setup.sh && chmod +x setup.sh  
&& bash setup.sh --update
```

Proxmox community Script

Go into the LXC and type in `update` <https://community-scripts.github.io/ProxmoxVE/scripts?id=patchmon>

Agents

Agents will auto-upgrade to 1.3.3 if the settings have been selected to allow this. Pinned release for the agent repo : <https://github.com/PatchMon/PatchMon-agent/releases/tag/1.3.3>

Many thanks to the community for their hard work and support. <3

<https://buymeacoffee.com/iby>

1.3.2

? Major Features

Docker Support (still in **beta**)

Previously the docker collector was a script that was ran (also through cron), now it's baked into the Agent binary file and therefore no need for a separate bash script. It also leverages the same `credentials.yml` which was introduced in 1.3.0. We have also added more information that is collected such as networks and volumes for a complete picture of your Docker environment.

Forced agent update button

You'll now find a button on the host page to force update the agent if it doesn't wish to update automatically.

UI themes to chose from

A few new branding Ui themes have been added allowing you to chose what theme to apply to PatchMons interface. This is currently app-wide and it requires that dark-mode is enabled for these themes to work.

Performance

Additional environment variables have been added relating to Database connections, these are all documented here : [Environment Documentation](#)

Metrics

We have introduced a metrics system, more information is found here about how our metrics collection works, what data is collected and opting in/out etc :

<https://docs.patchmon.net/books/patchmon-application-documentation/page/metrics-collection-information>

TFA / Backup Codes

Fixed TFA remember me not actually remembering Fixed Backup Codes entering, they can now be used in the same text box as the code itself

Fixes

1. Fixed Host timeout issue due to SSE connection issues on frontend

2. Fixed https go agent communication with server
3. Fixed Docker inventory collection
4. Fixed TFA and Backup Codes
5. Fixed not grouping by groups in the hosts table
6. IPv6 listening support added in Nginx config by community member @alan7000
7. When Deleting Groups it shows the hosts that are being affected

P.S I skipped 1.3.1 version tag because some members in the community have 1.3.1 when I was building it, if we release it as 1.3.1 then their agents won't really update properly - catering for the few.

Docker upgrade instructions video : https://www.youtube.com/watch?v=bi_r9aW2uQA **Written**

Instructions in docs : <https://docs.patchmon.net/books/patchmon-application-documentation/page/upgrading-from-130-to-132>

For bare-metal type the `curl -fsSL -o setup.sh`

`https://raw.githubusercontent.com/PatchMon/PatchMon/refs/heads/main/setup.sh && chmod +x setup.sh && bash setup.sh --update` *should update your instance*

Many thanks to the community for their hard work and support. <3 iby__

1.3.0 (Major)

? PatchMon version 1.3.0

This major release brings a new and improved architecture on the server and agent.

? Major Features

GO based Agent

Agent is now a GO based Agent binary file that runs as a service `systemctl status patchmon-agent`

The Agent serves a websocket connection to PatchMon server.

Agent has been compiled in amd64, i386, arm and arm64

A new repository has been setup for the agent.

BullMQ + Redis

The PatchMon Server runs a BullMQ service which utilises Redis server for scheduling automated and queued jobs. Jobs include things like "Cleanup orphaned repos" where it will remove repositories that are now not associated with any hosts etc Bullboard has also been added so that we can have a dashboard to monitor the jobs from a server level.

WebSocket

PatchMon Agents now connect via Web Socket Secure (https) or Web Socket (ws) to listen for commands from PatchMon. The Agents themselves control the schedule of reporting information however this persistent and bi-directional connection architecture lays the foundation of PatchMon so that it can control and handle management etc.

Performance

Various performance related improvements have been made with the way that node.js uses prisma for the Postgresql ORM. There was a lot of connection leakage where instead of utilising established connections it would create a new connection to the Database. These were causing at times Database connections to rise above 100! Fixes also improved the way the front-end speaks to the /api/v1 endpoints. These remove and handle the 429 (rate limit errors) and other backend errors.

Security

Various security handling has been improved around cookie handling, cors handling etc so that /bullboard can be authenticated

Agent updates checking

New mechanism for checking for Go based agents. The PatchMon server will query the GitHub repo and allow you to download the agents directly whilst the agents themselves will query PatchMon. I have pinned the agent version with the server version and had the agents query the server for downloading the updates as opposed to downloading them from github. This is because I plan to use PGP for signing agents off in the future and improve security mechanisms for server/agent verification.

Upgrading

Bash scripts from 1.2.8 will use an intermediary script of 1.2.9 which will run the installation of the new agent service. **Docker upgrade instructions video :**

<https://www.youtube.com/watch?v=NZE2pi6WxWM> **Written Instructions in docs :**

<https://docs.patchmon.net/books/patchmon-application-documentation/page/upgrading-from-128-to-130> **Coming soon:**

For bare-metal type the setup.sh update is being modified soon to handle the installation and setup or Redis 7 DB user and password as well as the nginx configuration amendments to handle upgrade on the websocket and add the /bullboard directive.

Many thanks to the community for their hard work and support. <3 iby___

1.2.8 to 1.3.0 - Upgrade

Upgrading the Server

Introduction

Upgrade Video link : <https://www.youtube.com/watch?v=NZE2pi6WxWM>

There are 3 main changes between version 1.2.X and 1.3.x:

1. **Go-based Agent Binary:** The introduction of a binary based on Go for the agent, replacing the previous bash scripts. This binary executes much faster and is compatible across different architectures when compiled.
2. **Redis and BullMQ Integration:** The introduction of Redis as a back-end database server and BullMQ as the queue manager for tasks and automation management.
3. **Nginx Configuration:** The addition of an nginx block for the presentation of the /bullboard URL.

Let's go through the two types of upgrades:

Docker Upgrade

This is quite simple as we just need to add the following in the container configuration for Redis:

1. Add the Redis service
2. Add the Redis configuration in the backend environment
3. Add a new redis_data volume

Important: Ensure you change the Redis password and update it in all three areas where "**your-redis-password-here**" is specified. This password should be secure but some **alphanumeric** characters can cause issues.

Docker Compose Ammendments

```
name: patchmon

services:
  redis:
```

```
image: redis:7-alpine
restart: unless-stopped
command: redis-server --requirepass your-redis-password-here # CHANGE THIS TO YOUR REDIS
PASSWORD
volumes:
  - redis_data:/data
healthcheck:
  test: ["CMD", "redis-cli", "--no-auth-warning", "-a", "your-redis-password-here",
"ping"] # CHANGE THIS TO YOUR REDIS PASSWORD
  interval: 3s
  timeout: 5s
  retries: 7

backend:
  environment:
    # Redis Configuration
    REDIS_HOST: redis
    REDIS_PORT: 6379
    REDIS_PASSWORD: your-redis-password-here # CHANGE THIS TO YOUR REDIS PASSWORD
    REDIS_DB: 0
    # ... other environment variables

volumes:
  redis_data:
```

Migration issues

If you get a migration issue like this:

```
backend-1 | Error: P3009
backend-1 |
backend-1 | migrate found failed migrations in the target database, new migrations will not
backend-1 | be applied. Read more about how to resolve migration issues in a production database:
backend-1 | https://pris.ly/d/migrate-resolve
backend-1 | The 20251005000000_add_user_sessions migration started at 2025-10-21
backend-1 | 22:50:32.244874 UTC failed
backend-1 |
backend-1 |
dependency failed to start: container patchmon-backend-1 is unhealthy
```

Then you need to apply the following commands from the directory where the docker-compose.yml file is:

Depending on your docker environment and version it may be as `docker compose run`

```
docker-compose run --rm backend npx prisma migrate resolve --rolled-back  
20251005000000_add_user_sessions
```

```
docker-compose run --rm backend npx prisma migrate resolve --applied  
20251005000000_add_user_sessions
```

Bare Metal / VM Upgrade

Instructions for bare metal and VM upgrades will be detailed in the following sections... soon... Still building the script to handle the update ...