

# Migrating from 1.4.2 to 2.0.0

## This is a migration document for those on 1.4.2 to 2.0.0

I've only tested this from 1.4.2 to 2.0.0 . If you're on a lower version like 1.3.x then i'm not sure this will entirely work but there is a chance it will as the main aspect is the database migration.

**Please backup your container or host in a way where you can easily restore if needed to rollback.**

Usually there are three types of deployments:

1. via docker-compose.yml
2. via our native setup.sh script
3. via proxmox community scripts

docker-compose.yml is the preferred and supported way.

We had setup.sh originally when we first started the project, at that time we didn't have docker images and we have carried on supporting it until now.

The issue with installing PatchMon natively on an OS is that there are evolving changes as we have progressed and keeping on top of them (also via edge cases) becomes a very heavy task.

Shipping containers is much easier and also more secure as now we are using hardened images from docker to really reduce the CVEs that images are shipped with.

Technically there is a way still to not use docker and that is by serving the Go based binary directly.

I will talk about it but I can't support it officially. I can only point to the right direction.

The proxmox-community scripts team are a great bunch and we speak to them regularly about these major changes and work with them so that upgrade / installation is done seamlessly. They pin the scripts to a specific version so when we release a new version then it takes a few days until the team test the script against the new version.

For now I will document the process for

1. Upgrading via docker
2. Conversion of native setup.sh type to docker

The main thing when it comes to migrations is carrying over the database and ensuring that the new .env has the necessary entries.

The new V2 brings new features like Patching but more importantly has been completely re-written in **GO LANG** . Both agent and server are now in GO.

We have seen memory footprint of PatchMon server go from 500MB to 50MB and execution times become **much faster**.

## Docker migration from V1.4.2 to V2

### Key changes made in new version :

1. Single container for "patchmon-server" as opposed to "backend" and "frontend" . The frontend built static files are now embedded in the backend go binary file served via the container
2. Agents volume is now not needed - binaries are now embedded in the container image as opposed to a writable volume.
3. Branding assets volume also now not needed - images are now stored in the database as opposed to a writable assets volume
4. Fully utilised the .env file for all variables and is shared between the various containers - this keeps the docker-compose.yml clean and we only need to work on looking at the single .env file for any variables.
5. Utilising docker hardened images upon build for additional security.

### Easy script migration for new docker-compose.yml and .env

1. Change your directory to where you have the docker-compose.yml and .env file
2. Run the following :

```
curl -fsSL https://raw.githubusercontent.com/PatchMon/PatchMon/refs/heads/main/tools/migrate1-4-2_to_2-0-0.sh | bash
```

This will download the new docker-compose.yml and .env whilst migrating your variables to it. Please review your new .env before performing docker compose up because if you had a non standard port mapped to the container then this will also need to be double-checked.

## Setup.sh bare-metal to Docker

I'm currently working on writing this up - if you have less than 10 hosts or so consider starting from fresh - it will be easier :)

## Proxmox Community Scripts

The community-scripts team will be working on a migration and installation path for the scripts and will work with your "update" as soon as it's available to.

---

Revision #8

Created 2026-03-17 14:22:10 UTC by lby

Updated 2026-04-24 21:48:53 UTC by lby