

Proxmox-NUT Homelab HOWTO — Step 6 : Update Debian / Take Snapshot / Upgrade VirtIO / Manage Snapshots

 chribonn.medium.com/proxmox-nut-homelab-howto-step-6-update-debian-take-snapshot-upgrade-virtio-manage-d80bfd1cf1a9

Alan C. Bonnici

October 22, 2024



Alan C. Bonnici

Maintenance and Snapshots

Another step has been added to the Proxmox-NUT HomeLab HowTo (PNHH) project, focusing on maintenance and introducing snapshots.

Password used in this HowTo: *1234..qwerty*



Update Debian

Proxmox automatically checks for updates by default, covering both Debian and itself. If you do nothing, the system will update itself. However, I prefer to perform periodic updates to my HomeLab setup. This approach allows me to ensure no events are missed and gives me the opportunity to restart the system periodically.

For production environments, the needs will differ. A stricter and more formal regime would be in place to monitor and manage Proxmox, along with its VMs and containers. Production environments may also have multiple nodes that replicate one another, allowing for a node to be taken down without impacting users. Others may achieve this through High Availability (HA) configurations. This discussion is outside the scope of this step.

As mentioned, I like to restart the entire setup. Therefore, before updating Proxmox, I shut down any VMs (and containers). Updating packages should not break anything (*Cloudstrike, anyone?*). Those with multiple nodes can upgrade a node, test and verify that everything works well, and then update other nodes. Those with a single node should back up the server if they (unwisely) perform critical tasks on one physical machine. As always, enterprise clients have additional support not available to non-paying users.

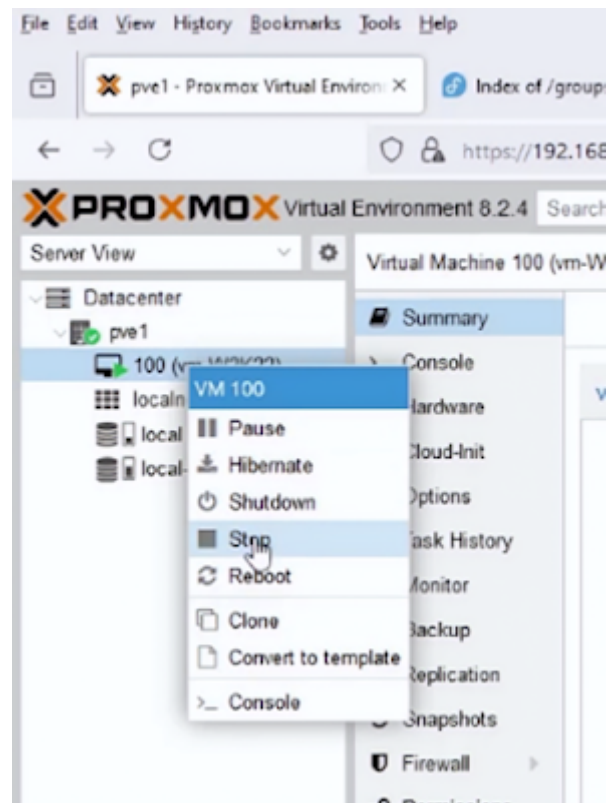
The update commands were discussed in another step of this series. They are:

apt update: Updates the package lists, including packages that need upgrading and new packages that have just been added to the repositories.

apt list -upgradable: Outputs a list of all installed packages that can be upgraded.

apt upgrade: Downloads and installs the package updates.

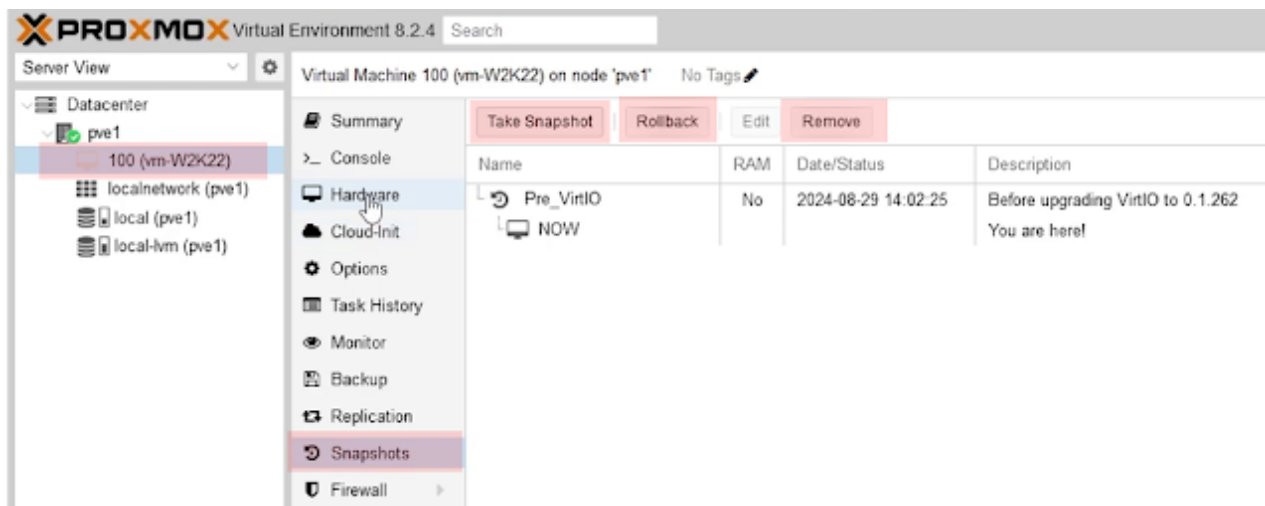
After the upgrade, you can issue a **reboot** command.



Snapshots

Not all storage types support snapshots. If the **Take Snapshot** button is greyed out, this means that your storage type does not support them. For a complete list of the various storage types and which ones allow Snapshots visit: <https://pve.proxmox.com/wiki/Storage>.

If you've downloaded the VMware images for this series, you're already using Snapshots. Snapshots, a term borrowed from photography, capture a moment in time for your virtual machine. They record the current state of the VM and track any subsequent changes. You can create snapshots while the VM is running or stopped. In this guide, I took the snapshot while the VM was stopped.



There are two main actions you can perform with snapshots:

- : This restores the VM to the state it was in when you took the snapshot. Any changes made after the snapshot are discarded.
- : This merges the changes made after the snapshot into the VM's current state and deletes the snapshot files.

It's important to note: Snapshots aren't independent copies of the VM. If the VM becomes corrupted, snapshots might be lost as well. Backups, on the other hand, are full copies of the VM, so you can restore it to a previous state even if the running VM encounters issues.

Backups offer more flexibility: They can be stored offsite and can be restored to a different machine, while snapshots are tied to the specific underlying VM they were created for. On the other hand, snapshots are faster to create because they only record metadata, not the entire VM image. The topic of Backups will be covered in a future Step in this series.

Update VirtIO Drivers (Windows)

NAVIGATION

→ [Home](#)

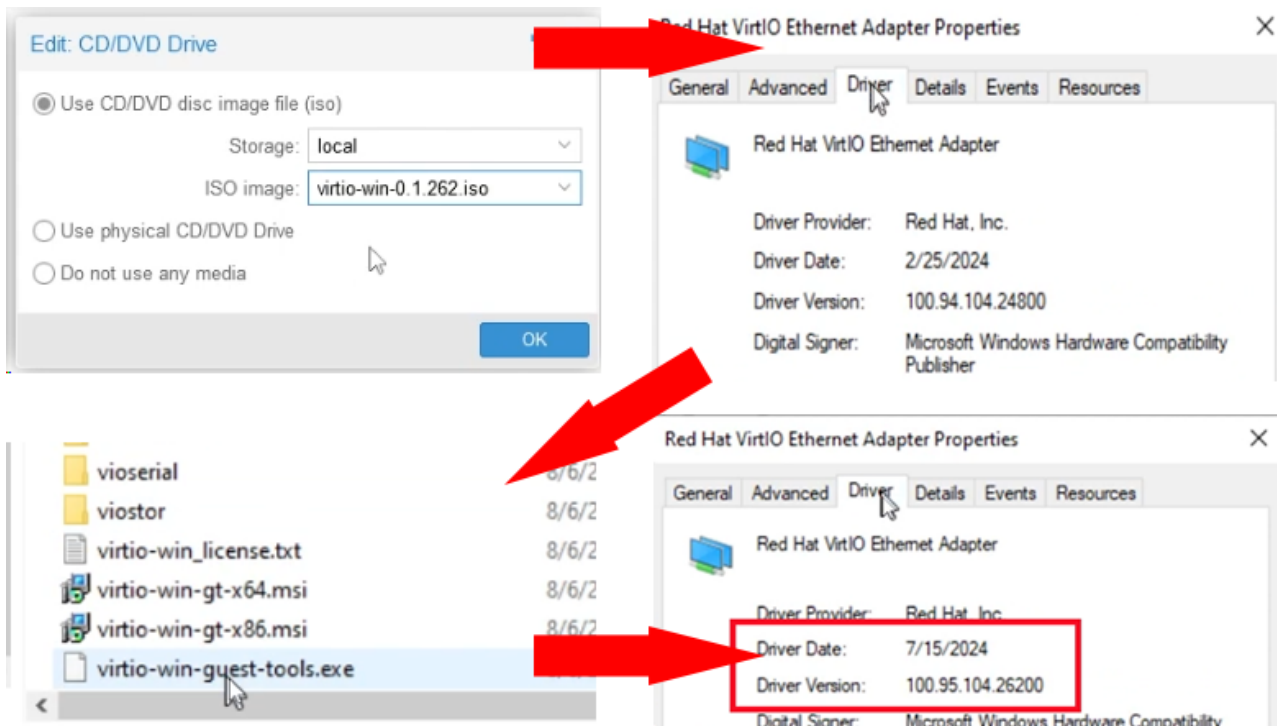
Name	Last modified	Size	Description
 Parent Directory		-	
 CHECKSUM	2024-08-06 03:26	0	
 archive-qemu-ga/	2024-08-06 03:29	-	
 archive-virtio/	2024-08-07 03:12	-	
 latest-qemu-ga/	2024-08-06 03:29	-	
 latest-virtio/	2024-08-07 03:14	-	
 stable-virtio/	2024-08-07 03:14	-	
 upstream-virtio/	2024-06-03 00:22	-	
 virtio-win-pkg-scripts-input/	2024-08-07 03:12	-	

We talked about VirtIO drivers in a previous step. Fedora has released a new version, but it seems they no longer provide a changelog to highlight the changes. While I found a forum discussion on Proxmox mentioning past changelogs, it appears this practice has stopped. In this case the latest and stable releases are identical. If, however, the versions differ, you'll need to choose the one that best suits your needs.

You can download the ISO from <https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/>.

The Steps

- 1. Create a Snapshot:** Before making any changes, take a snapshot of your VM. This ensures you can easily revert back if something goes wrong.
- 2. Mount the VirtIO ISO:** Insert the VirtIO ISO into your VM's CD drive.
- 3. Start/Log into the Windows VM:** If your VM is powered off, start it now.



4. Log into the VM:

- Note the current VirtIO driver version.*
- Run the VirtIO executable.*
- Reboot the VM.*
- After rebooting, log in again and confirm that the VirtIO drivers have been updated.*
- Verify that everything is functioning correctly.*

Cleanup/Finalize: Once you're satisfied that everything is working as expected, you can safely remove the snapshot. If there are issues and you can't resolve them, you can roll back to return the VM to its original state.