

# Timeshift

Timeshift is a backup and restore tool utilizing either `rsync` or `btrfs` snapshots.

## Installation

**WARNING:** Timeshift's `btrfs` mode is limited to an "Ubuntu-style" subvolume layout, i.e. only `@` and `@home` subvolumes can be present.

Install Timeshift from the AUR:

```
yay -S timeshift
```

## Configuration

### Cron

Timeshift relies on cronjobs to periodically create snapshots. Start the cron daemon via:

```
sudo systemctl enable --now cronie
```

Additionally, create an empty `crontab` file:

**ATTENTION:** If the `EDITOR` environment variable is not set `crontab` automatically calls `vi` to edit files. Either `export` it or supply it temporarily when calling `crontab`, e.g. `EDITOR=nano crontab -e` to use `nano` instead of `vi`.

```
sudo crontab -e
```

## Timeshift first time launch wizard

**ATTENTION:** The GUI wizard is buggy around `btrfs` and `lvm` setups. It will say no `btrfs` partitions were detected but lets you continue regardless if you select the device with the root `btrfs` partition.

Open Timeshift via the desktop launcher and complete the configuration wizard.

## Automatic snapshots on system changes

In addition to Timeshift's periodic snapshots, `timeshift-autosnap` provides a `pacman` hook to create a manual snapshot every time packages are installed, upgraded or removed. It works with either `rsync` or `btrfs` mode of Timeshift.

Install `timeshift-autosnap` from the AUR:

**TIP:** When using GRUB as your bootloader consider also installing `grub-btrfs` to include `btrfs` snapshots in GRUB boot options.

```
yay -S timeshift-autosnap
```

By default `timeshift-autosnap` only keeps 3 snapshots and deletes older ones. To change this, edit `/etc/timeshift-autosnap.conf` and either set `deleteSnapshots` to `false` or increase the number of `maxSnapshots`, for example:

```
skipAutosnap=false
deleteSnapshots=true
maxSnapshots=7
updateGrub=true
snapshotDescription={timeshift-autosnap} {created before upgrade}
```

## Prevent excessive snapshotting when using `yay`

By default, when installing or updating multiple packages from the AUR, `yay` first builds a package and immediately calls `pacman` to install it, before building and installing the next one on its list. This also means that the `timeshift-autosnap` hook is triggered **for each individual AUR package** built by `yay`, **including dependencies also installed from the AUR.**

This can have undesirable side-effects:

- `yay` will reach the `maxSnapshots` limit very quickly when installing multiple packages from the AUR, causing `timeshift-autosnap` to delete snapshots you may have wanted to keep
- if `deleteSnapshots` is set to `false` the available space on the `btrfs` partition may quickly be exhausted with an absurd amount of snapshots taken

To prevent this it is recommended to configure `yay` to:

1. not remove make dependencies after successfully built packages are installed
2. build all AUR packages first, install them all later
3. install AUR packages together with regular repo packages

By calling `yay` with the `--save` parameter, any options passed to it will be saved in a configuration file, e.g.:

```
yay --noremovemake --batchinstall --combinedupgrade --save
```

Next time you use `yay` to install, upgrade or remove packages it will read the generated config file at `~/.config/yay/config.json` and apply the options automatically without having to specify them during use.

---

Revision #4

Created 4 November 2021 16:18:27 by Sebin

Updated 26 March 2023 03:28:21 by Sebin