



Read-Only Raspberry Pi

Created by Phillip Burgess



<https://learn.adafruit.com/read-only-raspberry-pi>

Last updated on 2024-03-08 02:40:44 PM EST

Table of Contents

Overview

3

- [The Concern](#)
- [The Solution](#)
- [Before You Begin](#)
- [Enabling Read-Only /boot](#)
- [Restoring Read/Write Access](#)
- [You're Not Finished Yet](#)

Overview

Unlike most microcontroller projects that can simply be switched off when done, computers like the Raspberry Pi require an **orderly shutdown** procedure...otherwise the storage card may become **corrupted** and the system will no longer boot.

There are situations where just pulling the plug would certainly be handy, like slideshow or video kiosks, or systems left in the hands of non-technical users. Fortunately recent Raspberry Pi OS releases have a configuration option to make systems **more resistant to unplanned power cuts**.

The Concern

Linux—or any substantial computer operating system, Windows and Mac are the same way—behind the scenes they're frequently writing temporary data to disk (or the SD card with Raspberry Pi). This is why we normally use the **shutdown** command: every file is put away in a known valid state. But if power is unexpectedly cut, these lingering half-files can render a card **unbootable**...one can try patching it up, but often there's no recourse but to **wipe the card and reinstall everything**.

The Solution

Recent Raspberry Pi OS releases have an option to put the /boot partition (where the kernel image and other critical files reside) in a **read-only** mode. If software installed on this system only performs reading and playback operations (e.g. a [slideshow kiosk \(https://adafru.it/zrA\)](https://adafru.it/zrA), a [Fadecandy \(https://adafru.it/zrB\)](https://adafru.it/zrB) server, a [Halloween display \(https://adafru.it/zrC\)](https://adafru.it/zrC)), then you can just unplug the system when done. It's not ideal for read/write tasks like databases, web servers or data logging, but it has a place.

Before You Begin

- This works with **Raspberry Pi OS** and close derivatives (e.g. RetroPie), but might not work with other distributions.
- Setting up read-only mode should be the **very last step** before deploying a project. Get all your code and data on the system, get software auto-starting as needed, test it normally with the usual boot and shutdown methodology. It's easier up-front. Once you're 100% confident in its operation, then configure the system for read-only boot.

- **Back up** the contents of your SD card first. We've tested on a couple versions of Raspbian, but maybe something's changed, or has been overlooked, and could leave the Pi in a weird intermediate state.

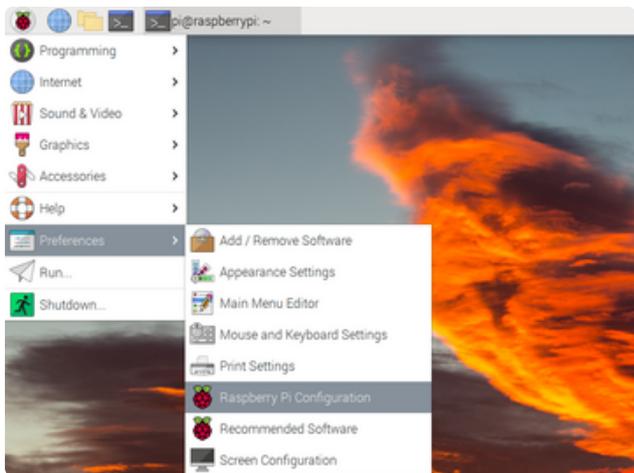
Enabling Read-Only /boot

We assume that you have a Pi booted and on a network...like mentioned above, everything already configured and fully functional (and backed up) before taking this step. We have [other guides \(https://adafru.it/dDL\)](https://adafru.it/dDL) covering the very basics of getting a Raspberry Pi started.

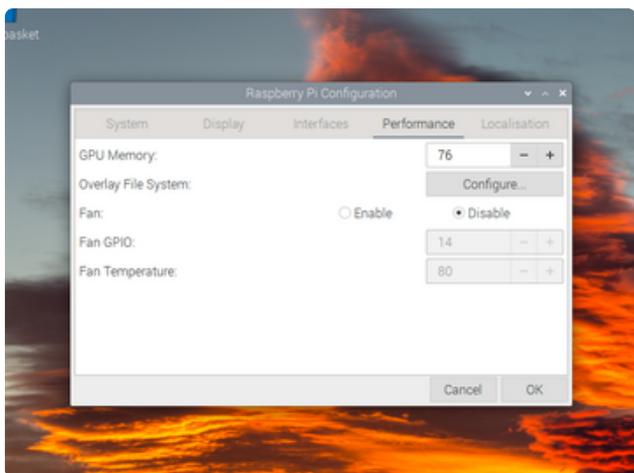
Using the Desktop/GUI “Full” OS

You'll find this setting in the Raspberry Pi Configuration tool. From the Raspberry menu at the top-left...

Pi→Preferences→Raspberry Pi Configuration



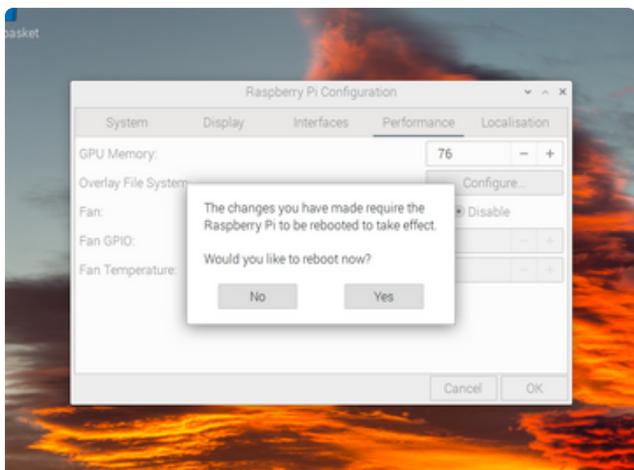
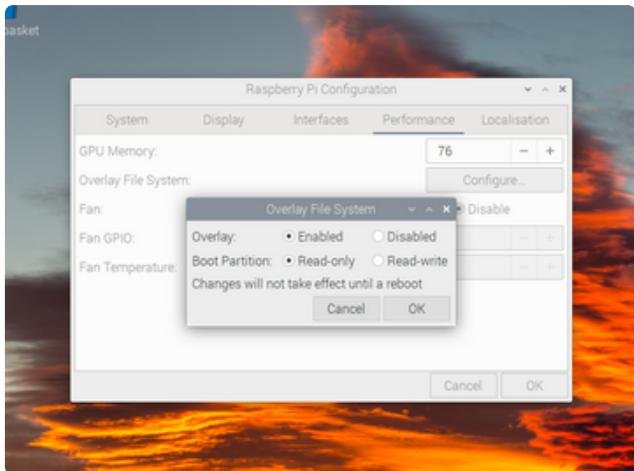
Go to the “Performance” tab and you’ll see an option “Overlay File System.” Click the “Configure...” button.



Select “Overlay: Enabled” and “Boot Partition: Read-only.”

Click “OK” and wait while the system works. It may take a minute or more to complete. This is normal.

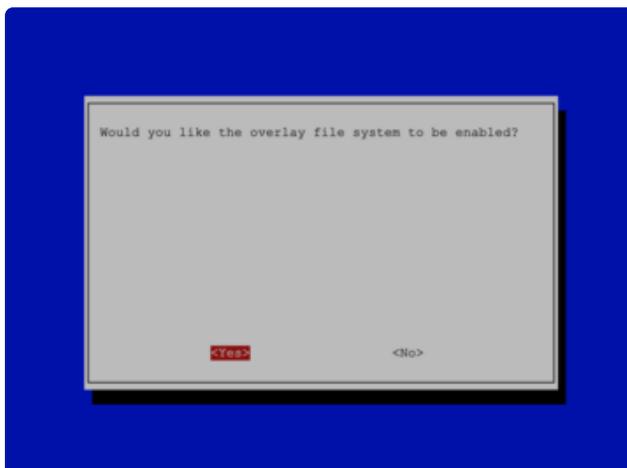
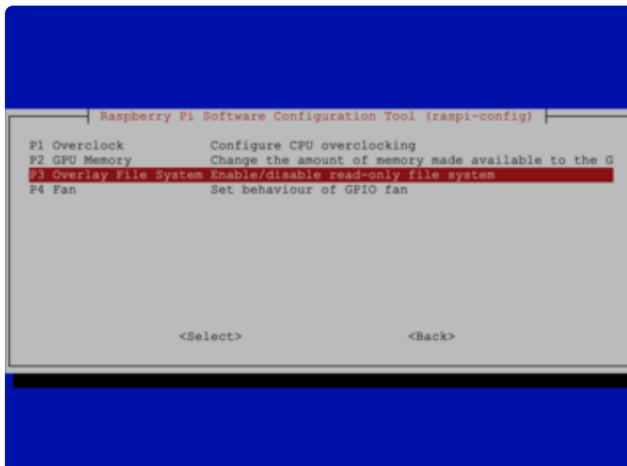
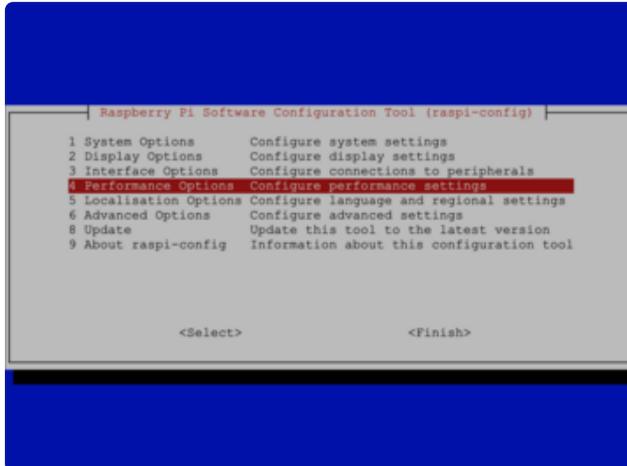
Reboot when prompted.



Using the “Lite” Command-Line OS

These options can be found in the `raspi-config` tool, which must be run as root:

```
sudo raspi-config
```



Navigate down to “Performance Options” and then “Overlay File System.” Select “Yes” to both the enable and write-protect questions.

It may take a minute or more while the system works, this is normal. Tab to the “Finish” button and reboot when prompted.



For Both

You're done! Now the /boot partition is safely locked. If you need to access something there though, like the config.txt file, or a major system update, instructions to temporarily enable write access are given below.

With each new OS release, it's normal that some configuration options may move to different menus or positions. If you can't find it where shown, check under the other top-level menu options...even if moved, the name will likely remain similar.

Restoring Read/Write Access

If you need to **temporarily** enable read/write access, as when editing tricky configuration settings in `/boot/config.txt`, or major system updates affecting the kernel or device tree files, this can be done from the command line (if using "full" Raspbian with a GUI, open a terminal window):

```
sudo mount -o remount,rw /boot
```

Perform your edits or upgrades, then reboot the system (via Pi→Shutdown with the GUI, or `sudo reboot` from the command line). It will be back in its read-only state.

If you need to **permanently** restore read/write access, you must first use the temporary step above, then can navigate to the same GUI or raspi-config settings, selecting "Read-write" for the GUI option, or "No" for the raspi-config write-protect question. Then reboot and the system's back to normal, with permanent read/write access.

If you have enabled read/write access to /boot, whether temporarily or permanently, always wrap up your session with an orderly shutdown or reboot.

You're Not Finished Yet

Test the modified system extensively to make sure that the system boots and your application runs as intended. If it doesn't...does the software you're using rely on temporary files in the /boot partition? Maybe it can be adjusted.

Now make an image of the SD card (using dd or Apple Pi Baker or whatever your backup tool of preference) and, if it's a critical application, **burn at least one spare**. There are other ways cards can go bad...static, brown-outs, falling out and getting lost...this read-only setup won't always save you. SD cards are cheap now! Spares help if you've left a system in someone else's care (let's say a museum kiosk) and it fails for some reason, you can ask them to just swap out the card until you can get there to troubleshoot. I know at least one Burning Man project rendered useless in the first few minutes of the event because their one and only card fell out and was lost on the playa.