User Manual

How to configure Raspberry Pi OS “Read-Only” mode by using “automatic-preparation” package.

Detailed instructions.

- The assumption is that on your workstation machine you are running one of the latest Ubuntu or Debian distributions and your shell is bash. If this is not the case, you will need to modify the scripts to match your distribution.
- Unpack the package and copy script files to a local directory. You can place them in the same directory where the Raspberry Pi OS zip file is located.
- Generate modified Raspberry Pi OS image.
  - Depending on your intention, along with read-only system configuration additional customization can be done by adding appropriate “**custom-additions-*.sh**” files to the package:
    - For basic read-only setup you would need just two files to locate in your working directory:
      - **prepare-system-image-for-read-only-mode.sh** - main script to run.
      - **prepare-system-image-for-read-only-mode-functions-00.sh** - functions for the main script.
    - To additionally limit system partition default resizing to the whole SD card, add “**custom-additions-01-limit-system-partition-resize.sh**” to the package.
      - This addition will set the limit for root partition resize by value that is twice greater than the value of the original system image before expansion.
      -
  - Open a terminal for the directory of the package location.
  - In the terminal run the main script by the command:
    - **`sudo ./prepare-system-image-for-read-only-mode.sh <directory_where_Raspberry_Pi_OS_zip_file_is_located>`**
      - Notes:
        - The script will automatically find and process files from the provided directory using naming scheme “**YYYY-MM-DD-raspios-*-armhf*.zip**”. Multiple image files can be processed at the same time.
        - the parameter of the script can be omitted if image zip files are located in the same directory with the script.
  - All operation will be performed in the **/tmp** directory.
    - In case of success:
      - intermediate temporary directories and files will be automatically deleted;
      - resulting image file(s) will be available in the **/tmp** directory for burning onto an SD card.
    - In case of failure the following directories with current content may be kept for troubleshooting:
      - “rpi-image---XXXXXXXXXXXX”,
      - “rpi-mount-root---XXXXXXXXXXXX”,
      - “rpi-mount-boot---XXXXXXXXXXXX”,
        - where “XXXXXXXXXXXX” are randomly generated strings,
      -
  - Note. All scripts mentioned below will be automatically generated and added to the Raspberry Pi OS Desktop image by running “**prepare-system-image-for-read-only-**

mode.sh" and will be available in the directory **/usr/local/sbin/** of your Raspberry Pi system.
    -
-
- Burn the image onto an SD card.
    - ~~~~~
-
- Insert the SD card into your Raspberry Pi board and start it.
    - … Booting to GUI...
    - Wait for Desktop with Welcome dialog.
- Open a Terminal.
- Run script
    - **`sudo conf-0-1[-...] <TAB> <ENTER>`**
    - (to collect Initial reports)
        - Full script name: "**conf-0-1-collect-initial-boot-reports.sh**".
-
- **In Welcome dialog**:
    - NOTE. *For Lite system use "sudo raspi-config" command instead*.
    - ~~~~~~~~~~~~~~~~ Welcome Dialog ~~~~~~~~~
    - **Page 0 "Welcome"**
        - Click button "Next"
    -
    - **Page 1 "Set Country"**
        - Select "Country"
            - *United States*
        - Select "Language"
            - *American English*
        - Select "Timezone"
            - *New York*
        - Select check box "Use English language"
        - Select check box "Use US keyboard"
        - Click button "Next"
    -
    - **Page 2 "Set Password"**
        - Enter your password twice
        - Click button "Next"
    -
    - **Page 3 "Set up Screen"**
        - Select the checkbox there if needed.
        - Click button "Next"
    -
    - **<u>Set up WiFi connection</u> (IMPORTANT** for UFW install and system update on the next steps**)**
        -
        - **Page 4 "Select WiFi Network"**
            - Select WiFi Network
                - Select your WiFi Network from the provided list by SSID.
            - Click button "Next"
        -
        - **Page 5 "Enter WiFi Password"**
            - Enter the password.
            - Click button "Next"
        -

- **Page 6 “Update Software” (We recommend to SKIP this step here)**
  - We recommend to **SKIP** this step in the dialog.
    - System update will be done a few steps away by a dedicated script using CLI commands. It will provide much more detailed information of what is going on during the update, not just a blind progress bar. It makes sense because the update process loads and installs more than a hundred packages and takes about 20 minutes.
  - Click button **“Skip”**
- **Page 7 “Setup Complete”**
  -
  - Click button **“Done”**
- ~~~~~~~~~~~~~~~~ END of Welcome Dialog ~~~~~~~~~~~~~~~~
-

- Run script to install UFW.
  - **`sudo conf-0-2[-...] <TAB> <ENTER>`**
  - (Install UFW and reboot in 10 sec)
    - Full script name: “**conf-0-2-ufw-install-enable-and-reboot.sh**”.
  - If succeeds, automatically reboots in 10 sec…
  -
- … Booting to GUI...
- Wait for Desktop.
- Open a Terminal.
-
- Run script to make full system update/upgrade.
  - **`sudo conf-0-3[-...] <TAB> <ENTER>`**
    - (Apt Update/Upgrade)
      - Full script name: “**conf-0-3-update-system-software-and-reboot.sh**”.
    - Loads and installs about 77+ packages of about 191+ MB as on Jun 30, 2021.
  - Wait about 10-20 min observing progress…
  - When done, it will automatically reboot in 10 sec.
-
- … Booting to GUI...
- Wait for Desktop.
- Open a Terminal.
- Run script
  - **`sudo conf-0-4[-...] <TAB> <ENTER>`**
  - (to collect reports after system update)
    - Full script name: “**conf-0-4-collect-reports-after-system-upgrade.sh**”.
-
- Run script
  - **`sudo conf-0-5[-...] <TAB> <ENTER>`**
  - (to prepare for read-only mode)
    - Full script name: “**conf-0-5-prepare-system-read-only-mode.sh**”.
  - For the question “**Do you want to switch the system to 'Boot to CLI' mode right now?**” answer “**Y**” (recommended).
    - By this command the following changes in the system are expected:
      - system command “raspi-config nonint do_boot_behaviour B2” will be called and among other things (if any) the following changes will be done in the /etc/systemd/system directory:
        - existing systemd default.target symlink will be deleted;
        - new systemd default.target symlink will be created pointing to multi-user.target;
    - Alternatively, you can skip this for now and make it later, on the next step.

    - If succeeds, automatically reboots in 10 sec…
-
- … Booting to CLI (or, to GUI, depending on what you have chosen)...
- Wait for CLI prompt (or Desktop; open a Terminal if in desktop).
- .
- Run script
    - **`sudo conf-0-6[-...] <TAB> <ENTER>`**
    - (to permanently switch the system to boot in read-only mode)
        - Full script name: "**conf-0-6-switch-permanently-to-read-only-mode-and-reboot.sh**".
    - If succeeds, automatically reboots in 10 sec…
    -
- … Booting to CLI in read-only mode...
- Wait for CLI prompt.
-
- Run script
    - **`sudo conf-0-7[-...] <TAB> <ENTER>`**
    - (to collect reports in read-only mode)
        - Full script name: "**conf-0-7-collect-reports-after-switching-to-read-only-mode-and-reboot.sh**".
    - If succeeds, automatically reboots in 10 sec…
    -
- ~~~~~~
- The above steps take about 40-60 minutes.
- ~~~~~~
- Now, **your RPi is configured to run in read-only mode**.
- ~~~~~~

License.

This library is supplementary material to the article "How to configure Raspberry Pi OS to run in read-only mode".
The article and this library can be found at https://altomaxtech.com/rpi-ro/.

Email us at support@altomaxtech.com.

~~~~~~
~~~~~~