Base Headless Pi Setup

This page describes how to get a Pi to a base level headless configuration (ie, SSH & networking enabled, all packages up to date)

- *Minimum sd card size is 4GB*

WARNING: Old method of Pi SD card setup NO LONGER WORKS!

Used to be that you could just write the Raspbian image to a card, add the `ssh` file in the `/boot` partition, and ssh into the pi, but the normal `pi` user was removed in new OS versions. Instead, the Pi Imager tool must be used.

Install the imaging software and write the SD card

1. install the imaging software. On arch:

   ```
   sudo pacman -S rpi-imager
   ```

   (On other OS's, the package manager is different, but the package name should be the same or similar.)

2. Plug in the SD card that will go into the pi
3. Open the "Raspberry Pi Imager" software and choose an image ("Choose OS" button --> Raspberry Pi OS (other) --> OS Lite)
4. Hit the gear icon to change these settings:
   - hostname
   - enable ssh
   - username/password
   - wifi (if the pi has wifi)
   - Locale (time zone/keyboard layout)
5. Choose the SD card as the Storage device
6. Click the "Write" button

Boot the Pi and login

1. Insert the SD card into the Pi
2. plug in ethernet to the Pi if not using wifi (most USB ethernet adapters are supported)
3. Apply power to the Pi
4. wait for it to come online and log in via SSH (check local DHCP server logs for Pi's IP)

Pi Zero 1/2 settings not applying:

It seems the Pi Zero (or the legacy 32-bit OS images) don't play nice with the Pi-imager software, and so the settings you put in there (eg. hostname, user name/password, SSH settings) ARE NOT APPLIED to the SD card! You have to manually set up SSH and a default user BEFORE starting the Pi!

1. Insert the newly-flashed SD card into your PC, mount the boot (`rootfs`) partition.
2. Make 2 new files: `ssh` (empty file, no extension), and `userconf.txt`
3. edit `userconf.txt`, add the line:

   ```
   username:hashed_password
   ```

   where `username` is your chosen user name, and `hashed_password` is your securely-hashed chosen password, which can be generated by this linux terminal command:

   ```
   openssl passwd -6
   ```

   Enter your chosen password, and the output is the hash to enter into the `.txt` file above! The contents of an example file, with the user name `dev` and a hashed password, looks like:

   ```
   dev:$6$qiMOrC8lupyYGFcI$16fEo0XdZLjfsslYvsLgMmjZMhdrn8.HfDkp/UezCvKFIppGMLhgLbhHL4VwB7LU3b1WmGcmTmv9HGO1B0y3J0
   ```

Now you can boot the Pi! Note that both of these files are deleted automatically after the first boot - they are used to enable the SSH server, create your user account, then are promtly removed. This is the "old method" of creating headless Pi's. If the Pi Imager app is fixed such that its advanced options work again, then that method is easier, so do that instead!

Post-install steps (after logging in):

1. run `sudo raspi-config` to finish SD card setup:
   - 6 Advanced Options --> A1 Expand File System
2. Set the hostname in the System Options menu
3. Set the locale and time zone in the Localisation Options menu

4. Enable i2c under Interface Options (if required for your project)
5. Reboot the Pi
6. Update packages with `sudo apt-get update && sudo apt-get upgrade`

The Pi can now be used for it's intended project.

Add SSH Keys

To allow login from Yubikey or other private keys, public keys should be added to `authorized_keys` file in the Pi user's home directory. Downloading keys requires internet connection:

1. `mkdir ~/.ssh`
2. `curl https://github.com/your-github-username.keys > ~/.ssh/authorized_keys`
3. `chmod 700 ~/.ssh`
4. `chmod 600 ~/.ssh/authorized_keys`
5. (Optional) disable password login: edit `/etc/ssh/sshd_config` and add/modify the password auth lines to:

```
PasswordAuthentication no
PermitEmptyPasswords no
PubkeyAuthentication yes
RSAAuthentication yes
```

Reboot Pi after changing SSH config.

Revision #14
Created 2022-06-11 15:59:31 UTC by Dev
Updated 2025-12-27 22:07:38 UTC by Dev