

Running ChurchInfo is not complicated if you have experience with Web applications. If you don't, there are a few things to get used to. It is important to understand is that ChurchInfo is a Web-based application, which means it has two distinct sides:

The "server", on which the application actually runs. This is your Raspberry-Pi that stores the files and information that ChurchInfo needs to run.

The "client", through which a user interacts with the application via a Web browser.

There is only one server, but there can be an unlimited number of clients.

1) Churchinfo-Pi is a turnkey image of ChurchInfo already installed into a mysql database including a lightweight (php-enabled) webserver (hiawatha) running on top of a Linux operating system ready to be transferred to an SD Card, plugged into a Raspberry Pi connected to your existing Local Area Church ethernet network (your church LAN) and run. The whole idea is to simplify the installation and setup process as much as possible while keeping the cost low for onsite hosting (all of your congregation's personal information remains onsite.)

2) The downloaded image, churchinfo-pi-0.6-beta.img (once is unzipped) needs to be transferred to your 8 or 16 GB Class 10 SD card (4 GB should work also depending on your database size.) The unzipped decompressed images will be under 2 GB which should leave plenty of room for your data once the last partition and filesystem is expanded to the size of the remaining space of the SD card.

The following discussion provides the tools to transfer the .img to the SD card using a Windows or MAC computer.

**For flashing the SD card using a Windows PC or Mac refer to the the following Web site:**

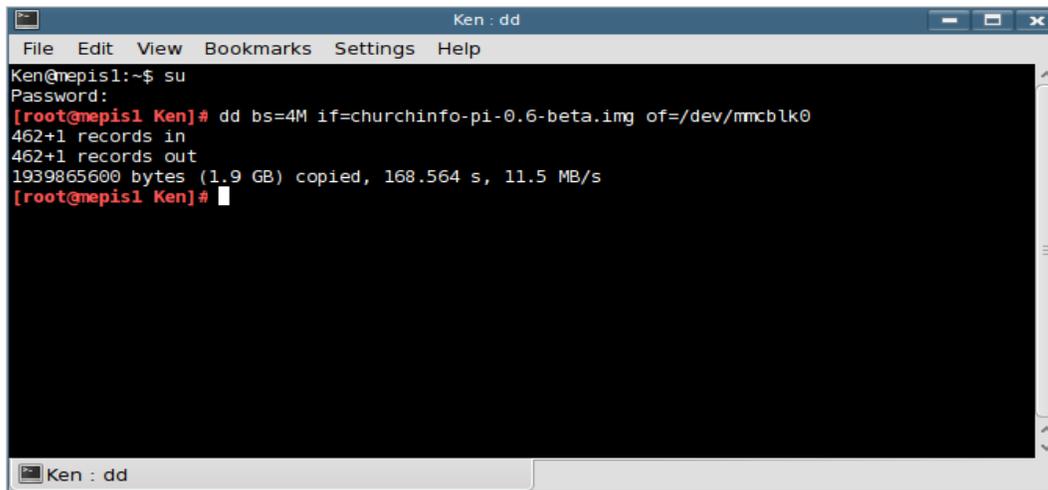
[http://elinux.org/RPi\\_Easy\\_SD\\_Card\\_Setup](http://elinux.org/RPi_Easy_SD_Card_Setup)

Go to section 3.2, Flashing the SD Card using Windows or section 3.3, Flashing the SD card using Mac OSX, Substitute " churchinfo-pi-0.6-beta.img" (minus the quotes) for the Raspbian .img file name.

**Note: Use the following method for Linux systems.**

**Flashing the SD card on a Linux OS:**

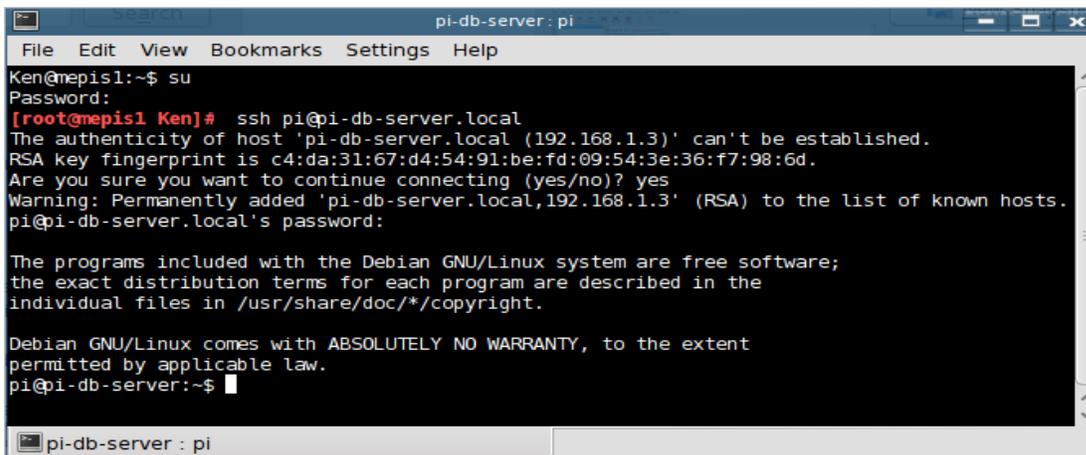
- Format the SD card using a tool such as Gparted. **Make a note of the SD card identification. Caution: Gparted and the use of the dd command will overwrite any partition on your computer. Be sure you have selected the SD card.**
- Format the SD card to an **ext4 format**. (Contrary to some sites, the SD card cannot be a FAT format.)
- Open the terminal program supplied with your OS and select the directory where your .img file is located.
- Enter the following command:
- **sudo dd bs=4M if=churchinfo-pi-0.6-beta.img of=/dev/sdd (replace sdd with the SD card id identification noted above, do not include the partition number if shown)** There is no progress output during the transfer, so be patient, it takes about 3 + minutes on a dual core 64 bit machine.



```
Ken : dd
File Edit View Bookmarks Settings Help
Ken@mepis1:~$ su
Password:
[root@mepis1 Ken]# dd bs=4M if=churchinfo-pi-0.6-beta.img of=/dev/mmcblk0
462+1 records in
462+1 records out
1939865600 bytes (1.9 GB) copied, 168.564 s, 11.5 MB/s
[root@mepis1 Ken]#
```

- Once the image has been properly transferred to the SD card (and the card is unmounted), remove the card and install it in the Raspberry Pi (already powered off.) With the Raspberry Pi (assuming Model B) already connected to your LAN via Cat 5e Ethernet and the power cable connected, plug in the power cable to the outlet and the boot up process will commence (leave the audio / video ports disconnected -- no need for these.)
- Avahi / Zeroconf is now broadcasting the server's hostname, which is pi-db-server, over your LAN. So, you should be able to use that, appending ".local" in lieu of the IP address. To proceed with the initial setup process login to the Raspberry Pi via ssh (Windows users may use putty download link at <http://www.putty.org/>, Linux/Mac users open a terminal session and enter:

**ssh pi@pi-db-server.local** (ssh is already enabled.)



```
pi-db-server : pi
File Edit View Bookmarks Settings Help
Ken@mepis1:~$ su
Password:
[root@mepis1 Ken]# ssh pi@pi-db-server.local
The authenticity of host 'pi-db-server.local (192.168.1.3)' can't be established.
RSA key fingerprint is c4:da:31:67:d4:54:91:be:fd:09:54:3e:36:f7:98:6d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'pi-db-server.local,192.168.1.3' (RSA) to the list of known hosts.
pi@pi-db-server.local's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
pi@pi-db-server:~$
```

Log-in to your Linux OS (you can enter su and your root password, or on some Linux systems you can only log\_in using the sudo command.)

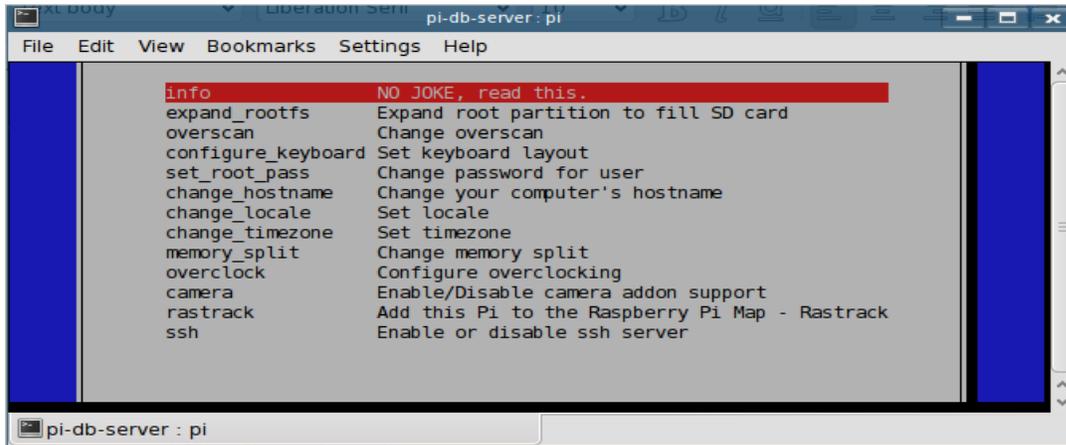
**pi admin user password is churchinfo777**

Upon successful log-in, enter:

**sudo server-setup** (you must use sudo at this point even if your Linux OS allows you to log-in as root)

Enter the password: **churchinfo777**

Which will launch the initial setup and configuration application and you will be presented with the following screen:



The most important option to select is `expand_rootfs` (next line down from the highlighted bar) to expand your filesystem to make use of all available space on your SD Card. Select `<Finish>` (use the right arrow to select) and reboot the pi.

Log back into the pi-db-server with:

```
ssh pi@pi-db-server.local
```

password: **churchinfo777**

Restart the server-setup program:

```
sudo server-setup
```

password: **churchinfo777**

Next select the **change\_timezone** (If you are in the US it is not necessary to select `change_locale` as default locale is set at `english_us`, the default timezone is set at `US/Central`.) Note, the time zone locations are displayed in a small area with just a few locations shown, use the down arrow to view more zones.

Note:

Currently the root account is locked (and it is recommended to leave it that way -- use `sudo` for safe server administration. If you set a new root password via the server-setup program, it will likely re-enable the root login account.) You may change the hostname, however be sure to change how you access the server then as `ssh pi@[new_hostname].local` and as `http://[new_hostname].local` (from the clients.)

Also, the `memory_split` is pre-set at the minimum value of 16M for the GPU and leaving the maximum remaining available memory for the CPU/processor. So, it is not recommended that you change that at all for best results (as the GPU is not needed.)

Once you complete the initial setup options exit the server-setup program, **login to the Raspberry Pi once again via ssh to change the "pi" administrative user's password.** Once login is complete, enter `passwd` (it will then ask for your new password twice -- for confirmation.) Be sure to remember this new password and/or store a copy in a secure location.)

Once complete, enter `sudo shutdown -r now` (which will initiate the reboot process.)

Exit the terminal program.

4) You should now be able to access ChurchInfo via your browser at:

**http://pi-db-server.local/churchinfo** (provided you did not change the hostname in the previous step.)

The initial user is "**Admin**" with a password of:

**churchinfoadmin** (passwords are case insensitive)

You will be prompted to change this password upon login. Once you have created other user accounts, you may delete or rename this default account. Just make sure that you always have a user with administrative privileges.

5.) Select Admin->Edit General Settings and Admin->Edit Report Settings to finish customizing your installation. See the Help menu for suggestions for setup.

6.) When there is a need to shutdown the Raspberry Pi server, be sure to login via ssh (using putty for Windows users), and enter `sudo shutdown -h now` before pulling the plug (or otherwise cutting the power), otherwise there is a HIGH risk of filesystem corruption which could cause data loss.)

Passwords: user: pi          churchinfo777  
              mysql root        mysqlpiroot1212  
              mysql user: pi    churchdborg777