The Heart of It All: Building a Media Center with Raspberry Pi

In this article by **Thomas Hamilton**, the author of Building a Media Center with Raspberry Pi ,you will learn how to find the operating system that you will use on the system that you chose. Just like with hardware, there are a plethora of options for the operating systems for the Raspberry Pi. For this book, we are going to focus on transforming the Raspberry Pi into a media center. At the time of writing this book, there are two operating systems available that are well known for being geared specifically to do just this. The first one is called the **Open Embedded Linux Entertainment Center** (**openELEC**) and is a slimmed-down operating system that has been optimized to be a media center and nothing else. The second option, and the one that we will be using for this project, is called the **Open Source Media Center** (**OSMC**). The main advantage of this specific version is that there is a full operating system running in the background. This will be important for some of the add-ons to work correctly. Once you can do this, if you want to try openELEC, you will be fully prepared to be able to do this on your own. In fact, the information in this article will enable you to install practically any operating system that's designed for a Raspberry Pi onto an SD card for you to use and experiment with as you see fit. In this article, we will cover the following topics:

- Downloading an operating system
- Installing an operating system to an SD card using Windows
- Install an operating system to an SD card using Linux
-

(For more resources related to this topic, see here.)

The Operating System

It is now time to find the correct version of OSMC so that we can download and install it. If you are primarily a Windows or an iOS user, it may feel strange to think that you can search online for operating systems and just download them to your computer. In the Linux world, the world in which the Raspberry Pi resides, this is very normal and one of the great things about open source. The Raspberry Pi is built as a learning tool. It was designed in such a way that it will

allow you to modify and add to it. In this way, the community can develop it and make it better. Open source software does the same thing. If you know programming, you can contribute to and change software that someone else developed, and this is encouraged! More eyes on the code means less bugs and vulnerabilities. Most versions of Linux follow this open source principle.

Versions of Linux? Yes. This is another point of confusion for Windows and Mac users. For the computers that you buy in a normal retail or computer store, you do not have many choices related to the OS that is already installed. You can either buy an Apple product with the newest version of their OS, or a Windows-based computer with Windows 7, 8, or 10 pre-installed. In this example, Windows 7, 8, and 10 are just newer and older versions of each other. Linux works off a different principle. Linux itself is not an operating system. Think of it more like a type of operating system or maybe as a brand such as Microsoft and Apple. Because it is open source and free, developers can take it and turn it into whatever they need it to be. The most popular versions of Linux are Ubuntu, Fedora, Suse, Mint, and CentOS. They each have a different look and feel and can have different functions. They are also operating systems that can be used daily for your normal computing needs. This article is based on a combination of Ubuntu and Fedora operating systems. The world of Linux and open source software can be confusing at first. Don't be scared! After you get past the shock, you will find that this openness is very exciting and helpful and can actually make your life much easier. Now, lets download OSMC.

Raspberrypi.org

If you haven't come across this already, it is the official website for the Raspberry Pi. From this website, you can find information about the Raspberry Pi, instructional how-tos and forums to talk with other Raspberry Pi users. This site can point you to their official retailers for the versions of the Raspberry Pi that are currently in production, and for the purpose of this article, it points us to the most popular operating systems for the Raspberry Pi (though not nearly all the ones that can work on it). From the main page, click on the link that says **DOWNLOADS** near the top of the page. This will bring you to the page that lists the most popular operating systems. Raspbian is the official OS of the Raspberry Pi and what OSMC is based on. Noobs is worth looking at for your next project. It isn't an OS itself, but it gives you the ability to choose from a list of operating systems and install them with a single click. If you want to see what the Raspberry Pi is capable of, start with Noobs. Under these options, you will have a list of third-party operating systems. The names may sound familiar at this point, as we have mentioned most of them already. This list is where you will find OSMC. Click on its link to go to their website.

We could have gone straight to this website to download OSMC, but this allowed you to see what other options are available and which is the easiest place to find them. OSMC gives a few different ways to install the OS onto different types of computers. If you want to use their automated way of installing OSMC to an SD card for the Raspberry Pi, you are welcome to do so; just follow their instructions for the operation system that you are using on your main computer. For learning purposes, I am going to explain the method of downloading a disk image and doing it ourselves, as this is how most operating systems are installed to the Raspberry Pi. Under the heading named **Get Started**, where you can choose the automated installation methods, there is a line just under it that allows you to download disk images. This is what we are going to do. Click on that link.

Now, we are presented with choices, namely Raspberry Pi 1 and Raspberry Pi 2. The Raspberry Pi 1 refers to any of the single-core Raspberry Pi devices while the Raspberry Pi 2 refers to the newest Pi with a quad-core processor and more RAM. Click on the link under whichever heading applies for the type of Pi that you will be using and select the newest release option that is available.

Verifying the download

While OSMC is downloading, let's take a minute to understand what the MD5 Checksum is. An MD5 Checksum is used to verify a file's integrity. The number that you see beside the download is the Checksum that was created when the file that you are downloading was created. After the image has finished downloading, we will check the MD5 Checksum of the file on your computer as well. These numbers should be identical. If they are not, it indicates that the image is corrupt and you will need to download it again. From a security standpoint, a checksum can also be used to ensure that data hasn't been tampered with in the time span between when it was created and when it was given to you. This could indicate malicious software or a data breech.

Now that OSMC has been downloaded, we can verify its integrity. In Linux, this is easy. Open a terminal and navigate to the *Downloads* folder or wherever you downloaded the file. Now type in the following command:

```
[md5sum name-of-file]
```

The output that this gives should match the MD5 Checksum that was beside the file that you clicked on to download. If it doesn't, delete the file and try doing this again.

To verify the file integrity using Windows, you will need to install a program that can do this. Search online for MD5 checksum Windows, and you will see that Microsoft has a program that can be downloaded from their website. Once you download and install it, it will work in a fashion that's similar to the Linux method, where you use the Windows command prompt. It comes with a readme file to explain how to use it.

If you are unable to find a program to verify the checksum, do not worry. This step isn't required, but it helps you troubleshoot whether the Raspberry Pi will not boot after you install the OS onto the SD card.

Installing OSMC - for Windows users

For Windows, you need to install two more applications to successfully write OSMC to an SD card. Because the OSMC file that you downloaded is compressed using gzip, you need a program that can unzip it. The recommended program for all of your compression needs in Windows is WinRAR. It is free and can be found at www.filehippo.com along with the next program that you will need. After you unzip the OSMC file, you will need a program that can write (burn) it to your SD card. There are many options to choose from, and these options can be

found under the **CD/DVD** option of **Categories** on the homepage. ImgBurn and DeepBurner appear to be the most popular image burning software at the time of writing this article.

Preparing everything

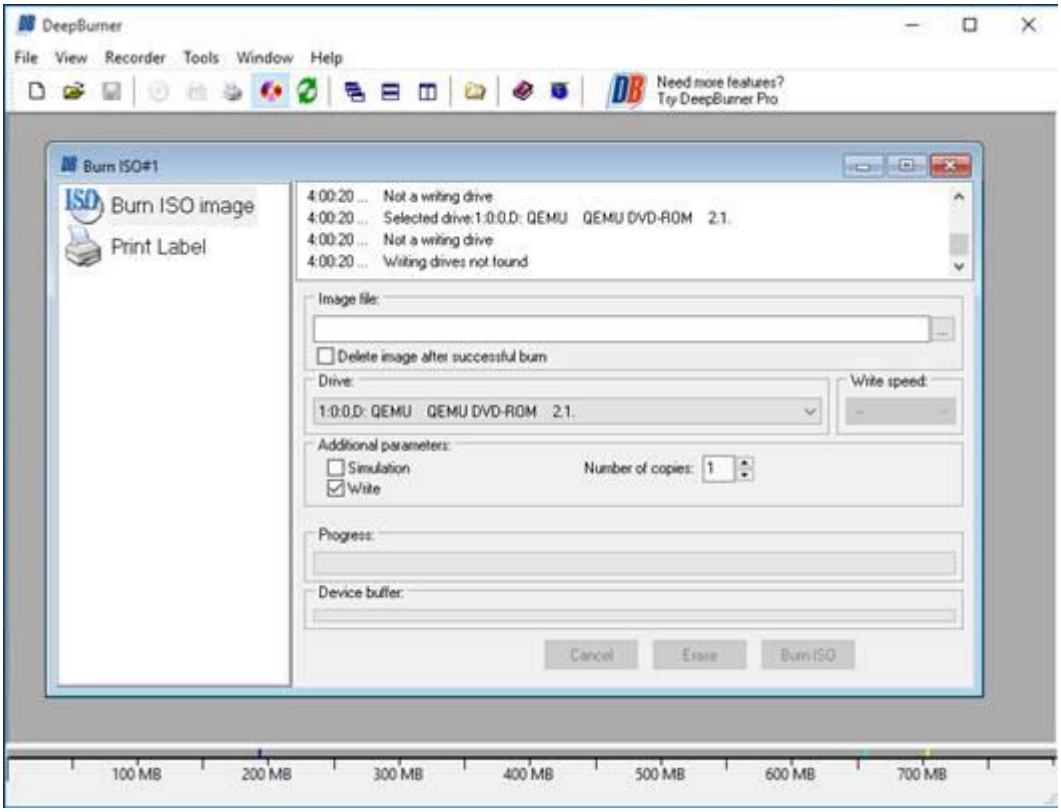
Ensure that you have the appropriate type of SD card for the Raspberry Pi that you own. The original Raspberry Pi Model A and B use full-size SD cards. Thus, if you purchased a miniSD by mistake, do not worry. The miniSD probably came with an adapter that turns it into a full-size SD. If it did not, they are easy to acquire.

You will need to insert your SD card into your computer so that you can write the operating system on it. If your computer has an in-built SD card reader, then that is ideal. If it does not, there are card readers available that plug in through your USB port and which can accomplish this goal as well.

Once you have inserted your SD card into your computer using either method, ensure that you have taken all the information off the card that you want to keep. Anything that's currently on the card will be erased in the following steps!

Install WinRAR and your image burning program if you have not already done so. When it is , you should be able to right-click on the OSMC file that you downloaded and select the option to uncompress or extract the files in a gzip file.

Burn It!



Now that we have an OSMC file that ends with *.img*, we can open the image burning program. Each program works differently, but you want to set the destination (where the image will be burned) as your SD card and the source (or input file) as the OSMC image. Once these settings are correct, click on **BurnISO** to begin burning the image. Now that this is done, congratulations!

Installing OSMC - for Linux users

As you have seen several times already, Linux comes with nearly everything that you need already installed. The software used to install the operating system to the SD card is no different. Ensure that you have the appropriate type of the SD card for the Raspberry Pi that you own. The original Raspberry Pi Model A and B use full-size SD cards. Therefore, if you purchased a miniSD by mistake, do not worry. The miniSD probably came with an adapter that turns it into a full-size SD. If it did not, they are easy to acquire.

Preparing the SD card

You will need to insert your SD card into your computer so that you can write the operating system on it. If your computer has an in-built SD card reader, then that is ideal. If it does not, there are card readers available that plug in through your USB port that can accomplish this goal as well.

Once you have inserted your SD card into your computer using either method, ensure that you have taken all information that you want to keep off the card. Anything that's currently on the card will be erased in the next step! If the SD card was already formatted with a filesystem, it probably automounted itself somewhere so that you can access it. We need to unmount it so that the system is not actually using it, but it is still inserted into the computer. To do this, type the following command into your command line:

```
lsblk
```

This command lists the block devices that are currently on your computer. In other words, it shows the storage devices and the partitions on them. Sda is most likely your hard drive; you can tell by the size of the device in the right columns. Sda1 and sda2 are the partitions on the sda device. Look for your device by its size. If you have a 4 GB SD card, then you will see something like this:

```
NAME                      MAJ:MIN           RM        SIZE    RO TYPE
MOUNTPOINT

sda                           8:0           0     238.5G        0       disk

├─sda1                        8:1           0     476M          0       part
/boot

└─sda2                        8:2           0      186.3G       0       part
/

sdb                           8:16          1      3.8G         0       disk

├─sdb1                        8:17          1      2.5G         0       part

└─sdb2                        8:18          1      1.3G         0       part
/run/media/username/mountpoint
```

In this case, my SD card is sdb and the second partition is mounted. To unmount this, we are going to issue the following command in the terminal again:

```
sudo umount /dev/sdb*
```

It will then ask you for your *sudo* (administrator) password and then unmount all the partitions for the sdb device. In this case, you could have replaced the sdb* with the partition number (sdb2) to be more specific if you only wanted to unmount one partition and not the entire device. In this example, we will erase everything on the device so that we unmount everything. Now, we can write the operating system to the SD card.

Burn It!

The process of installing an OSMC to the SD card is called burning an image. The process of burning an image is done with a program called *dd*, and it is done via the terminal. *dd* is a very useful tool that's used to copy disks and partitions to other disks or partitions or to images and

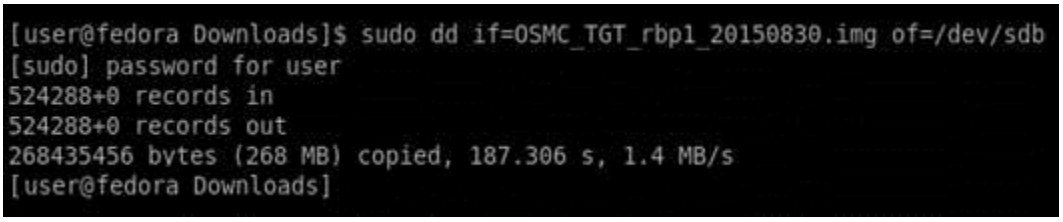
vice versa. In this instance, we will take an image and copy it to a disk. In the terminal, navigate to the directory where you downloaded OSMC. The file that you downloaded is compressed using gzip. Before we can burn it to the disk, we need to unzip it. To do so, type in the following command:

```
gunzip name-of-file.img.gz
```

This will leave you with a new file that has the same name but with the *.gz* file no longer at the end. This file is also much bigger than the gzipped version. This *.img* (image) file is what we will burn to the SD card. In the previous step, we found out what device our SD card was listed under (it was sdb in the preceding example) and unmounted it. Now, we are going to use the following command to burn the image:

```
sudo dd if=name-of-file.img of=/dev/sdb  (change /dev/sdb to whatever it is
on your computer)
```

And that's it! This will take several minutes to complete and the terminal will look like it froze, but this is because it is working. When it is done, the prompt will come back and you can remove the SD card:



Summary

If your computer already uses Linux, these steps will be a little bit faster because you already have the needed software. For Windows users, hunting for the right software and installing it will take some time. Just have patience and know that the exciting part is just around the corner. Now that we have downloaded OSMC, verified the download, prepared the SD card, and burned OSMC on it, the hardest part is over.