

Media Servers on non-RPi by Roller Angel



Media Servers are wonderful additions to a network. They allow all kinds of player devices to find them and stream the media off the media server.

What does it take to turn a standard FreeBSD RELEASE system into a fully functional media center? It's actually pretty straightforward. First, you must decide what server you will be installing. There are a few I can recommend. Plex Media Server would be my first choice as its fully-featured interface is a joy to use. To install it, you only need to type `sudo pkg install -y plexmediaserver`. Plex does offer a Plex Pass for a fee that allows additional features and there is a different command to use if you have that, which is `sudo pkg install -y plexmediaserver-plexpass`.

The next choice for a decent media server is Emby. The command for that is `sudo pkg install -y emby-server`. Finally, you can just install a mini DLNA server. Since Plex includes its own built-in DLNA server in addition to its own server software, I have mainly just stuck to using that DLNA server. However, those looking for something simple can just type `sudo pkg install -y minidlina`. Many players have built-in support for locating and streaming media from DLNA servers. It's important to read the information displayed after installing a package. To pull up that information again, type `pkg info -D` and the name of the package. If you want the media server to automatically start when FreeBSD is started, make sure you add an entry in your `/etc/rc.conf`.

Configuration

After installation you need to do some configuration. This is mainly just telling the server software where it ought to look for the different media types on your hard drive. Provide the software with the folders, and it will do a scan to not only bring the files into the system so they can be served,



but it will also try to match any additional information about the files with data from the Internet. For instance, if one of your files was the movie *Tron: Legacy* in the Movies folder named *tron.mkv*, Plex would see that, and instead of showing that filename, the listing for *Tron* in the Plex interface is lovely and filled with movie posters, actor names, descriptions, and many more features. Plex and Emby have additional applications that can be installed on mobile phones and smart TVs to allow one to access their media servers from these devices as well and not just from the web interface.

I'll walk you through configuring Plex Media Server as it's the one I use most often. With other media servers, the process is very similar. By the way, I'm running Plex on the same AMD64 desktop machine from the January/February 2019 *FreeBSD Journal* article I wrote called *A Guide To Getting Started with FreeBSD*. Once installed, I usually like Plex to start up on boot so I type `sudo sysrc plexmediaserver_plexpass_enable=YES`. As you can see, I have a Plex Pass. I mainly use it, so I can sync media from my server to my Android when I want access to media offline. I also find the movie trailer feature handy as I can quickly watch a trailer to see if I feel like watching a particular movie that I haven't seen before.

Next I need to start up Plex, so I type `sudo service plexmediaserver_plexpass start`. Now that Plex is running, I open the following URL in a web browser on the same network as my media server; keep in mind my IP is probably different from yours, but the port and path should be the same—`172.16.28.2:32400/web`. Once on that page, I just need to sign into my Plex account, and I can tell Plex that I like to keep my music type of media in the `/mediacenter/Music` dataset, movie type of media in the `/mediacenter/Movies` dataset, and the TV show type of media in the `/mediacenter/Tv Shows` dataset. Once I've set up a few types of media and finished the welcome wizard, Plex will scan all the media and match any media it can with media it knows about from the Internet. If Plex guesses wrong, I can un-match and use the interface to see what similar matches it can find, or I can simply provide my own content like posters, titles, and descriptions.

For the older TVs in the house, I have a Google Chromecast plugged in. I can use the Plex application on my Android phone to cast any of the media files to the TV. For my newest TV, the Samsung Smart TV, I can just go to the app store on the TV and install Plex to use it directly with the remote control to stream any media file. When I drive to work, I have my music synced to my Android so I can just play that media in offline mode to save data while not on WiFi. If mobile data limits aren't an issue, consider setting up OpenVPN at the house so that a remote connection to your media server can easily be made over the VPN from the device. If you're not keen on setting up a VPN, Plex does offer to allow media access outside of the network and has plenty of good information online for setting it up. I prefer to use a remote access VPN to access machines at home.

Jails

Consider running your media server in a jail. Michael W Lucas has a wonderful new book for helping master jails, <https://mwl.io/nonfiction/os#fmjail>, and it



explains all the benefits and features of jails. There's also a good tip in there that should be implemented on media servers. It has to do with speeding up Python on FreeBSD. Apparently, mounting `/dev/fd` in your `/etc/fstab` speeds up Python. Great! All you do is add an entry into your `/etc/fstab` that looks like the following:

```
fdescfs /dev/fd fdescfs rw 0 0
```

ZFS Datasets

Also consider how you will be working with your files from the media server. Is the delete option in the media server interface important? Then the application will need to have permission to read and write to your media files. You may be able to get away with having a server with only read access to the media files if you want to do it that way. Consider using ZFS datasets for each media type. With separate datasets, you can experiment with ZFS to see how optimized you can make the storage depending on the typical media attributes. A movie file is usually a big file; audio files aren't usually large, but this can obviously vary. And it's the same with photos. Knowing the typical file size will help you when tuning ZFS datasets. See the FreeBSD Handbook for a great resource on ZFS and dig deeper with ZFSbook.com. •

Roller Angel is an avid BSD user who enjoys all the amazing things that can be done with BSD technology. He has taught programming workshops based on FreeBSD and is working on building an online training platform for teaching BSD and related technologies. See BSD.pw for more information.



LISA19

LISA: Where systems engineering and operations professionals share real-world knowledge about designing, building, and maintaining the critical systems of our interconnected world.

October 28–30, 2019 | Portland, OR, USA
www.usenix.org/lisa19

Registration and the program will be available in August 2019.
Sign up to learn more: www.usenix.org/lisa/freebsd
and follow @LISAconference on Twitter.

