

# Streaming Video to your Android Phone

This document describes how to stream video to your Android phone via the local network or over the Internet.

## 1. Introduction

More and more of us have largely digital media libraries and would like to be able to view them wherever we are on whatever we want. An Android phone is a great tool for this purpose, but there isn't a straightforward way of achieving this seemingly mundane goal.

## 2. Creating the Files

Perhaps the easiest method is to use HandBrake (<http://handbrake.fr/>) - an open source video converter that is, sadly, fixated on the Apple market. Luckily, anything Apple can do we can do better!

Except get a stable Linux version. Sadly the Ubuntu/Fedora builds haven't been updated when Gnome made some changes so you're stuck with unstable releases. First you'll need to add the repository, then you can install Handbrake. Just run these commands to get the installation going:

```
sudo add-apt-repository ppa:handbrake-ubuntu/ppa
sudo apt-get update
sudo apt-get install handbrake-cli handbrake-gtk handbrake-common
```

You should now have HandBrake available under Applications --> Sound & Video --> HandBrake. The default settings seem to work fine as they're MP4-based, but you'll need to check the *Web optimized* button. It's up to you if you want to use the standard `.mp4` extension or the Apple-proprietary `.m4v` - the proprietary extension lets me know I've converted the file so I'll batch change them once I'm done.

Speaking of batch, HandBrake is pretty poor at batch file conversion. Soon I hope to digest this thread (<http://forum.handbrake.fr/viewtopic.php?f=10&t=5070>) to get a good CLI conversion script but I'm still spot checking.

I've also had issues with ripping a DVD directly (I've tried a whopping one DVD so far - and it was fairly modern). You may want to look into other conversion utilities as well, recently I've come to like

the multi-platform MakeMKV (<http://www.makemkv.com/>) which will get you non-DRM MKV files to play with. I may also play more with settings for DVDFab (<http://www.dvdfab.com/>) which I own and AnyDVD (<http://www.slysoft.com/en/>) is their competitor. Both are Windows-only so you'll need a Virtual Machine (see below) to run them. DVDFab's default MP4 output is not "Web optimized" as HandBrake puts it, but I'll need to explore the settings now that I have a baseline working solution.

This section obviously needs some updating, but I'll leave that for another day. I'll assume you've now got some trusty MP4 files living somewhere that you want to push out to your Android phone (or web browser, for that matter). Note that I haven't optimized any of the settings for the DroidX, nor for bandwidth constraints so some level of tweaking may be desirable.

More details on encoding video for Android (for local playback) can be found here (<https://help.ubuntu.com/community/AndroidVideoEncoding>).

## **3. Installing the Media Server**

I've taken a look at quite a few media servers, so far TVersity (<http://tversity.com/>) seems to be the best option. Not only is it a great DLNA ([http://en.wikipedia.org/wiki/Digital\\_Living\\_Network\\_Alliance](http://en.wikipedia.org/wiki/Digital_Living_Network_Alliance)) server, but it also sports a shiny web interface for streaming video. OK, it's pretty darn ugly but it's bandwidth efficient. Not very useful for a server that's going to be pushing a ton of video data at us, but still...

The best part, they're promising a native Linux version (<http://forums.tversity.com/viewtopic.php?f=1&t=13993>). I'd urge you to post your support for this in the thread and, should it work well, consider purchasing the Pro version.

### **3.1. Other Options**

Some of the other solutions I've tried include MediaTomb, which seems to work well as a native Linux DLNA server but has no means of streaming video over the web or to non-DLNA clients (which, sadly, the X must be considered as the DLNA client is far too specific in formats to be useful), and Jinzora which has a web interface and even an Android client - but sadly it doesn't seem to work very well.

### **3.2. Installing Windows**

Sadly, TVersity doesn't work under Wine nor Crossover Linux (<http://www.codeweavers.com/products/cxlinux/>). So you'll have to give up on Linux and install Windows. Just kidding, you'll need to dual boot and only watch video when Windows is running. Or you can always buy a dedicated Windows machine.

Or, you can download VirtualBox (<http://www.virtualbox.org/>) and set up a Virtual Machine. Proper VM management is outside the scope of this document, but it's pretty easy. VirtualBox has both a wizard-driven GUI as well as a robust set of command line tools. Once you've got it installed, just follow the defaults for a Windows installation of whatever flavor you can - however I usually bump up the memory to about 1G, depending on how much memory your physical machine has.

Something you'll need to change is to move the network interface from "NAT" (the default) to "Bridged". This will give you an IP address accessible from any computer on your network - more on that later.

Of course, always use licensed software, never search Torrent sites for things like "slipstream XP ISO", and make sure that your EULA allows you to install the Windows that probably came with your machine onto a Virtual Machine provided you remove it from the host machine. Of course, if you fail to follow these instructions I'm not liable for Microsoft's wrath.

Once the install is complete, install the VirtualBox Guest Additions. Run Windows Update a few times to get all the security patches and you should be all set. And you should also be much more appreciative of running **sudo apt-get update && sudo apt-get upgrade** to bring a Ubuntu box up to snuff after a fresh install. I also generally install Chrome on my VMs that have a GUI - but that's optional.

### **3.3. Installing and Configuring TVersity**

TVersity is a standard Windows install - download (<http://tversity.com/download/>) and run. Feel free to play with the free version, I'm holding off on upgrading to Pro until they release the Linux version.

Most of the default settings should work fine. If you want to access the server from outside your house you'll want to go to the General section, check the box that says "The media server should accept requests originating from outside the home network" or something similar. You'll also probably want to do some basic things like adding a username/password, and changing the port number couldn't hurt.

Now just add the files to your library. You'll probably want them on your Linux box, so verify that the Guest Additions are loaded and click on Devices --> Shared Folders to create a permanent share to the video folder (use your own judgment regarding whether the VM needs more access than that, or if it needs write access - I try to use read-only access whenever possible).

### **3.4. Configuring your Network**

If you want to stream to your phone when you're on Wifi, or to other devices like an Xbox or DLNA media player, you're done now. If you want to stream to your phone over 3G/4G you've got a little more work. You'll need to first figure out what IP address your virtual machine is, click Start --> Run and enter "cmd". Now type **ipconfig** and you should see your IP address (probably 192.168.x.y). Remember this number.

Now you'll need to log into your router which is generally found by changing the last number of the IP address to a 1. If you don't know how to configure your router, or you don't have access to it, now is a good time to Google the router model and find a manual or a forum. Check for something called Applications or Port Forwarding or something similar. You'll want to configure this so incoming requests to the TVersity port number (defaults to 41952) go to the IP address of the Windows XP virtual machine.

If you don't have one already, you'll also probably want to sign up for a free Dynamic DNS service. You only need one for all the computers and virtual machines on your network and I recommend Free DNS (<http://freedns.afraid.org/>) running on your Ubuntu box. This will let you go to a hostname instead of trying to remember your IP address and it will update for you should your IP address change.

The details of this section are beyond the scope of this document, but hopefully you know enough to Google port forwarding and your router model number to get a walk through.

## 4. Configuring Your Phone

You'll need a streaming media player for your phone as well. There are a lot of references to nswPlayer (<http://www.appbrain.com/app/com.nsw.android.mediaexplorer>), but I've also had good luck with Stream Media Player (<http://www.appbrain.com/app/com.psa.android.media>). Install one of these two and your phone is ready to go.

## 5. Playing Your Media

Let's start out easy. Open a web browser on your PC and point it to the Windows VM (we'll assume it's at 192.168.0.11). You'll want to include the port number in the URL so try <http://192.168.0.11:41952>. You should be prompted for a username/password if you set one otherwise you should see a TVersity page.

Once you've got that working, try it on your phone. With Froyo you can play with Chrome to Phone (<http://www.appbrain.com/app/com.google.android.apps.chrometophone>) and the associated Chrome Extension (<https://chrome.google.com/extensions/detail/oadboiipflhobonjffjbfekfjcgkhco>).

To try your external address you should disable Wifi, some routers can handle the packets coming in and out well and others can't.

## 6. Using DLNA to Stream to your Phone

Since the time I originally wrote this document, a new program called UPnPPlay (<http://www.appbrain.com/app/upnplay/cx.hoohol.silanoid>) has appeared on the market. This is a DLNA

client for your phone and seems to work a bit better than the built-in client on the Droid X, plus it supports multiple handsets. However you're still limited to the rather small number of playback codecs available for realtime streaming. Fortunately, VPlayer ([http://www.appbrain.com/app/vplayer-\(free-trial\)/me.abitno.vplayer](http://www.appbrain.com/app/vplayer-(free-trial)/me.abitno.vplayer)) (and the trial unlocker (<http://www.appbrain.com/app/vplayer-unlocker/me.abitno.vplayer.unlocker>) has come to the rescue. By installing both of these I've found that I can stream from my PlayOn (<http://www.playon.tv>) server to watch Hulu and Netflix (and others) and I can also use PS3 Media Server (<http://code.google.com/p/ps3mediaserver/>) (with the unfortunate acronym "PMS" ([http://en.wikipedia.org/wiki/Premenstrual\\_syndrome](http://en.wikipedia.org/wiki/Premenstrual_syndrome)")) to stream local content through a native Linux server.

This is a great way to stream your content to your phone and it seems to work very well in the limited testing I've done, but the major drawback to DLNA-based solutions is portability. This will only work if you're connected to your computer on the same network. You'll also need to make sure you can talk between your phone and your computer, some routers block this sort of communication by default as a security feature. Ideally you should also try to get your PC onto a hardwired Ethernet connection, that will help save your WiFi bandwidth a bit.

Far from a panacea, this is a reasonable (and reasonably-priced) solution to stream video in your own home. The PlayOn Android app (<http://www.appbrain.com/app/playon-mobile/com.playon.playonapp>) has been released, it essentially provides an app to navigate the web interface at <http://m.playon.tv/> (<http://m.playon.tv>). It uses Flash right now and, as such, requires Android 2.2. Here's hoping that the work with NetFlix, Comcast, Hulu, and FiOS will make these apps a bit easier to write in Honeycomb with less overhead and better rate adaptation.

To use the app or the browser interface, first install the PlayOn server, then connect your phone to a Wifi network co-located with your server. Then open up the PlayOn App (or the Android Browser and go to <http://m.playon.com>). You should be able to see your PlayOn server listed and play files from it while on Wifi. This action registers your phone to your server. Now when you open the app or go to <http://m.playon.com> while away from your Wifi network you should still be able to see your server and play the media. Not you'll need to check the PlayOn port settings and, if your router doesn't support uPNP well, you may be better off setting a fixed port and manually opening it up.

## A. About Me

My name is Jeff Bower, I'm a technology professional (<http://www.linkedin.com/in/jdbower>) with more years of experience in the telecommunications industry than I'd care to admit. I tend to post with the username jdbower on various forums, including Komodo Kamado (<http://komodokamado.com/forum/>), Android Central (<http://forum.androidcentral.com/>), VirtualBox (<http://forums.virtualbox.org/>), and MakeMKV (<http://www.makemkv.com/forum2/>). Writing these documents is a hobby of mine, I hope you find them useful and feel free to browse more at <https://www.ebower.com/docs>.

I also enjoy cooking, especially outdoors with my Komodo Kamado (<http://www.komodokamado.com>)

and using my Stoker (<https://www.rocksbarbque.com>). Take a look at my recipes stored at <https://www.ebower.com/recipes>.

If you've got any questions or feedback please feel free to email me at [docs@ebower.com](mailto:docs@ebower.com) (<mailto:docs@ebower.com>) or follow me on Google+ (<https://profiles.google.com/100268310848930740059>) or Twitter (<http://twitter.com/jdbower>).