

Droid X Hacking

The fantastic Android Community has created a ton of simple tools to help hack your Droid X. While sometimes documentation is lacking, no one thing really warrants a full document. However, there is some value in consolidating all of these tweaks in a single spot.

Of course, any of these tweaks is very dangerous and can result in doing bad things to your phone, needing to SBF back to stock, loss of data, voiding your warranty, hauntings from long-dead murder victims, zombie outbreaks, and ninja attacks.

1. SBF

"SBF" is one of those terms that can be used in a lot of contexts. An SBF file is essentially an image of your operating system, typically one of the stock Verizon loads so you can go back to factory fresh. SBF as a verb is to put this SBF file onto your phone using your computer. SBF doesn't have an agreed-upon acronym, I've heard System Boot Flash, System Binary File, and a myriad of other guesses. It also means Single Black Female, but that's a different website.

You'll need a program called `sbf_flash` (http://blog.opticaldelusion.org/search/label/sbf_flash). You can also download an older copy from me (`sbf_flash`) (don't forget to do a **`chmod +x sbf_flash`** before you use it). If you're running Windows or OSX, you'll also need an OS upgrade (<http://www.ubuntu.com/download>). In all seriousness, the Windows program RSD Lite seems to be very finicky and you may be better off with a Linux Live CD for this one.

You'll also need an SBF file. This is easier said than done, SBF files contain copyrighted material and therefore they move around a lot. Luckily, there is a great tool for finding Froyo (<http://lmgty.com/?q=droid+x+sbf+file+froyo>) and Gingerbread (<http://lmgty.com/?q=droid+x+sbf+file+gingerbread>) SBF files. For as long as it's up, this page (<https://sites.google.com/site/1kdsdroidx/home>) has been a good source and one of the filenames for the Froyo SBF is `VRZ_MB810_2.3.34_1FF_01.zip`, one for the Gingerbread is `VRZ_MB810_4.5.596_1FF_01.sbf.zip`. Note that Gingerbread, at the time of this writing, is not rootable. If you want to restore a borked stock install without root just SBF the Gingerbread image. If you want to run rooted Gingerbread, follow these instructions instead.

Since SBF files are large (about 500MB), chances are you've downloaded a compressed version. Uncompress it and make sure it's got an SBF extension.

Charge your phone, flashing an SBF can consume a lot of battery power and running out of juice in the

middle is a bad thing. Now power off your phone. Hold the Volume Down and Camera button. Now power on the phone. You should get a screen that has text like this:

```
Bootloader  
30.04
```

```
Battery OK  
OK to Program  
Connect USB  
Data Cable
```

This is good, and note that you don't need root or anything like that to get here. Now plug your phone into your computer and run the command **sudo sbf_flash [filename.sbf]**. After a bunch of validation, pushing, more validation, and general stuff happening you should be able to reboot your phone to a nice new version of your OS.

Note that to properly use your new SBF'd phone you may want to wipe your cache and perform a factory reset.

If you need any help, there's also an excellent tutorial here (<http://www.droidxforums.com/forum/droid-x-rs-guides/12015-complete-droid-x-sbf-flashing-guide.html>).

2. Wiping Cache and Factory Reset

Charge your phone, I can't tell you how often I see posts about "I thought 20% was enough to do this and now I'm stuck!" You'll also probably want to do a backup first since, in case you are a little slow, a factory reset will pretty much destroy most of any data and apps on your phone.

Power off your phone. Hold down the Home button and power up the phone - keeping the Home button pressed until you see a little Android with a big exclamation mark. Now here's where things get tricky, sometimes you'll need to hit the Search button, sometimes you'll need to press between the volume up/down button. Either way, you should get to a little menu overlaid on the exclamation. Use the volume keys to scroll to "wipe cache partition" and hit the power button to select it. Now scroll to "wipe data/factory reset" and use the power button to select.

3. Rooting Froyo

I rooted my phone back in the Eclair days using a manual process which did teach me a bit about how Android phones get rooted. Essentially when you install Linux on your computer you are both a user as well as the Administrator. When IT at work installs Linux, frequently you are a user and they are the administrator. This is so you don't accidentally do something stupid and screw up your system or their network. With your Android phone it's similar, the vast majority of people out there don't need root and really shouldn't root because you can screw up your phone (and then blame Verizon and make them give you a new one) or even screw up the network. However, phones can be rooted and some phones are easier than others.

The Droid X is pretty darn easy on Froyo. You'll need a program called z4root (<http://forum.xda-developers.com/showthread.php?t=833953>), you can download a potentially outdated version from me directly here (z4root.1.3.0.apk). Install the APK, charge your phone (you're not dumb enough to do these things with 10% battery life, are you?), click "Permanent Root", and just wait. Note you may need to enable USB debugging and unplug your phone for it to work properly.

4. Installing Rooted Gingerbread

There is not currently a known method for rooting the stock Gingerbread image, so getting a rooted Gingerbread is a little bit of work. Note that many ROMs (such as the current state of Liberty Gingerbread) require this rooted Gingerbread baseline install before you can install the ROM itself. Much of this information was gleaned here (<http://www.droid-life.com/2011/05/28/download-official-droidx-gingerbread-4-5-596-for-rooted-users/>).

First some prep work. You'll want to have some space on your SD card. You'll probably want a backup of your data (see Titanium Backup (<http://www.appbrain.com/app/titanium-backup-%E2%98%85-root/com.keramidas.TitaniumBackup>)). You'll want to download z4root (<http://forum.xda-developers.com/showthread.php?t=833953>) and the Droid 2 Recovery Bootstrap (<http://www.koushikdutta.com/2010/08/droid-x-recovery.html>) - note that you do NOT want the Droid X version for this. You can also download z4root (z4root.1.3.0.apk) and the Droid 2 Bootstrap (Droid2Bootstrap.apk) from me. Put these files on your SD card.

Now you'll need the GB release itself. You can download a copy of a torrent file here (DX+GB+596+All.torrent), or I'll refer you to the source here (<http://www.droid-life.com/2011/05/28/download-official-droidx-gingerbread-4-5-596-for-rooted-users/>). Inside you'll find files named `Blur_Version.2.3.340.MB810.Verizon.en.US-Part1.zip` and `Blur_Version.2.3.340.MB810.Verizon.en.US-Part2.zip`. These need to go to your SD Card as well.

You can also find similar files for the new 602 kernel of Gingerbread. You'll want to search for filenames `340-602_update_Part-1.zip` and `340-602_update_Part-2.zip`. Again, for copyright reasons I won't host them myself but the filenames and Google will get you where you need to go.

Next you'll want to SBF back to Froyo (I know, it hurts, but you won't be there long). Just to be sure, wipe cache and perform a factory reset. Don't bother logging into your Google account, we won't be here long and it'll slow things down.

Remember when I had you copy a bunch of files over to your SD card? Now they'll come in handy. Open up Files and browse to the z4root APK and install it. Now run it, permanent root, and your phone should reboot. Open up Files again and browse to the Droid 2 bootstrap APK and install it. Open it, click Bootstrap Recover and then Reboot Recovery.

Now you should have a text screen with ClockworkMod Recovery at the top and bottom. Use the volume buttons to scroll to "install zip from sdcard" and use the camera button to select. Scroll to "choose zip from sdcard" and then it's probably easier to scroll up to select the first Blur_Version... entry (note the filename is too long to show the entire file name and it *is* important to choose the right one). *Do not reboot!* Repeat this for the second file. Just to be safe, perform a factory reset.

You should now be able to reboot into a factory fresh, rooted Gingerbread environment. Be careful restoring apps, especially system apps. I've had much more luck with stability by manually reinstalling what I can and only selectively restoring data.

5. Installing CyanogenMod7

CyanogenMod7, or CM7, is a popular ROM in that it runs on just about anything. Now including the Droid X, at least in a pre-beta fashion. CM7 is a nice ROM, but there are a few things you need to know.

Unlike ROMs such as Liberty which are based off the Motorola official software, CM7 is built from source. This means that everything in CM7 is open source, and anything that's not open source doesn't work. The Blur camera app is pretty good, but won't run on CM7 without a bunch of force closes. HDMI support is gone. And as a new and very ambitious ROM there are a lot of bugs. But it's fun and workable as long as you're OK with what you're getting.

CM7 seems to live at RootzWiki and there are several good threads in the Droid X Dev Forum (<http://rootzwiki.com/forumdisplay.php?21-Droid-X-Developer-Forum>). Before you get started, you'll want to download a few files to your SD card. The latest CM7 release is here (http://cm-nightlies.appspot.com/?device=cdma_shadow) (and here (https://github.com/CyanogenMod/android_device_motorola_shadow/commits/gingerbread) is a list of Droid X specific changes). You'll want the apps listed in this thread (<http://rootzwiki.com/showthread.php?347-OFFICIAL-METHOD-HOW-TO-Prepare-Your-Phone-for-CM4DX!>), including the Droid 2 Bootstrap, z4root, and the Google Apps link. You'll also want razorloves' All-in-One fix (<http://rootzwiki.com/showthread.php?1347-All-In-One-Fix-Zip-for-CM4DX>) for some fixes (make sure you get the right AIO for the nightly build you're installing).

This will wipe your phone and all data on it, back up what you care about! You can follow the excellent

How-To (<http://rootzwiki.com/showthread.php?347-OFFICIAL-METHOD-HOW-TO-Prepare-Your-Phone-for-CM4DX!>) linked above or continue reading.

Now sbf back to Froyo (don't forget to wipe cache and data) and install z4root and the Droid 2 Bootstrap. Worried? Don't be, the CM7 build is based on the Froyo kernel but uses Gingerbread. Use z4root to root your phone, now use Droid 2 Bootstrap to Bootstrap Recover and then Reboot Recovery.

Once again, wipe cache and data. Now use the volume rocker to "install zip from sdcard" and use the camera button to select. Scroll to the CM7 install (should have a name like `cm_shadow_full-xx.zip`) and select it. Now wait. A while, it's a big file. Once it's done, install the Google Apps file (should be `gapps-gb-20110307-signed.zip` or similar). This should be much faster. Now do the same for the AIO fix (called `CM4DXfixn.zip`). Now reboot and you're good to go.

To upgrade to another CM7 build you'll need to install these three zip files again, but you don't need to wipe cache or anything like that.

Some quirks I've found:

- Alternate Keyboard Selection seems a little wonky. When I selected a keyboard it wouldn't hold the value. I don't know if this was because of the build I was on, but reinstalling build 33 seems to have resulting in the system coming back with the last keyboard I tried to enable checked. If you want to use Swype, for example, try enabling it in Settings --> Language & keyboard, if it fails to remain enabled install CM7, Gapps, and the AIO bundle without wiping any data and you should now be able to select it.
- Stock MotoSwype won't work, but you can sign up for the Swype Beta (<http://beta.swype.com/>) which works just fine.
- Using my Google Talk with Video installation method seems to have worked, but I needed to run the installer twice. Luckily, some previous instability I've had seems to have disappeared with nightly 34.
- HDMI support doesn't work. Even RealHDMI will be broken. CM7 doesn't have the basic drivers needed since they're proprietary and heavily embedded in the system. I wouldn't expect to see it working anytime in the near future.

In conclusion, CM7 is a decent ROM. I do like the toggles in the status bar dropdown and the extra items in the power off menu. It's more stable than any Rom freshly ported to the X should be (yes, there are a few bugs but also frequent fixes). But really, it's just an ADW-based ROM and I'm not so sure why it's so sought after. Not to belittle the herculean efforts that must have been exerted to shoehorn it into the X, but most ROMs give the ability to use the AOSP (Android Open Source Project - the stuff Google's released the code for) applications and I take a more pragmatic view of open source when it means losing capabilities like HDMI just because the drivers are proprietary. I've been on it for a few days now, and will likely stay with it for another week or two, but I expect to be headed back to Liberty eventually to regain HDMI support and the Blur Camera. I'm also curious about Justice, a CM7-based ROM that draws on Liberty components. Perhaps once CM7 and Justice settle down we'll see what proprietary things Jrummy is able to pull over.

6. Google Talk with Video

Some Gingerbread devices have access to Google Talk with Video, including the Droid X. Video chat is generally not all that useful on a mobile phone, either your arm will get tired holding the phone in front of your face or you'll give the infamous up-nose shot - not an appealing look! However, using the Droid X's rear facing camera can let you send a live feed of whatever you're looking at which is kind of cool. You can also rig up some mirrors to send video of yourself while being able to see the screen.

First, make sure you're on rooted Gingerbread or a Gingerbread ROM. Use something like Titanium Backup

(<http://www.appbrain.com/app/titanium-backup-%E2%98%85-root/com.keramidas.TitaniumBackup>) to backup your phone. This process will replace some key components and it's nice to be able to restore them. Note that I also lost my Google account settings when I did this, a minor inconvenience since I didn't lose my WiFi settings nor any apps because of it.

There is a zip file you can install with a bootstrap recovery program, but I haven't had much luck with it (installing it just leaves me with a stock Chat program, I haven't delved too deeply into the root cause of this but it may be a simple properties file I'm missing).

Thanks to this thread

(<http://android.modaco.com/content/software/339390/r1-12-may-google-video-talk-installer/>) you can download an installer, or you can download it from me here (com.modaco.googlecode.com/hg/googlevideotalk.apk). Simply run this file and it will ask for root permissions, update your Google Talk version and reboot your phone.

Once you're reconfigured your Google accounts you should now see camera icons next to some of your contacts (especially those who are logged in via the Gmail Talk interface). More deterministically you can also see "Voice and video chat settings" in the Settings menu.

7. Installing BASH

BASH is the world's most fantaburific shell ever. It's the default shell for most Linux distributions, supports tab completion for commands, robust history and scripting, and is generally just pretty darn cool. Someone at xda-developers named mzet has ported BASH to Android (<http://forum.xda-developers.com/showthread.php?t=537827>). Just for my sake, I've also stolen a copy for myself here ([bash](#)).

Note that some ROMs come with BASH built-in. You may have it in `/system/xbin/bash`. If you don't, here's how you can install it:

```
wget http://pub.mzet.net/bash
chmod +x bash
mount -o remount,rw /dev/block/mtdblock3 /system
mv bash /system/xbin/
```

```
mount -oremount,ro /dev/block/mtdblock3 /system
```

If you want BASH as your default shell you can run these commands - BUT THIS IS DANGEROUS SO BE CAREFUL!

```
mount -oremount,rw /dev/block/mtdblock3 /system
mv /system/bin/sh /system/bin/sh0
ln -s /system/sbin/bash /system/bin/sh
mount -oremount,ro /dev/block/mtdblock3 /system
```

To return to the original shell, just run the following:

```
mount -oremount,rw /dev/block/mtdblock3 /system
mv /system/bin/sh0 /system/bin/sh
mount -oremount,ro /dev/block/mtdblock3 /system
```

8. Installing Linux

Installing BASH not enough for you? You want the full Linux experience? Some people have gotten a full Linux chroot (<http://en.wikipedia.org/wiki/Chroot>) environment running on their X. Currently this is a stripped down version of Ubuntu, but the next iteration is moving toward Slackware-based Gentoo.

You'll need a few things to get started. First, I'm assuming you're rooted and probably running a custom ROM. The version of BusyBox with CM7 seems to work fine but some other versions are a bit broken according to legend (yes, I'm actually still running CM7 and getting used to the AOSP camera and lack of HDMI - plus I haven't had any time to go back to Liberty or try out Justice).

You'll also need to install the Android SDK (<http://developer.android.com/sdk/index.html>). I'm also going to assume you're running Linux, because if you want to run Linux on your phone and can't get it working on your computer you'll probably want to skip this tutorial. Note that you can run Linux in a Virtual Machine (I like VirtualBox (<http://www.virtualbox.org>)) and then just share the Android's USB port with the VM.

The full instructions are found here (<http://forums.androidclone.com/showthread.php?tid=23>) where you can also download Laika Beta 1.3 (http://www.filefactory.com/file/c1d0ed7/n/Laika_Beta_1.3.rar) (an Ubuntu-based distribution built for mobiles) or download it from me (Laika Beta 1.3.rar). REMEMBER: This page is for your Droid X. The page linked in this paragraph has a list of other phones supported by this process which may have slightly different implementations.

You'll also want the **bootubuntu** script, but there's a typo in it so I've fixed it and placed it here (bootubuntu). Place this file on your SD card before you begin.

If this is a fresh install of the Android SDK you'll probably want to run `~/android-sdk-linux_x86/tools/android update sdk` and download the suggested packages. All you really care about is making sure that `~/android-sdk-linux_x86/platform-tools/adb` exists. The following set of commands should help you install stuff on your phone. It will install into the internal storage area, if you want to install to your SD card, stop and ask yourself why. You've got gobs of space internally, but all of your media needs to sit in that tiny little 16GB (or 32GB if you've upgraded) SD card. Why waste it on a Linux install if you don't need to?

```
# Start from your home directory
cd ~
# Extract and mount the Ubuntu image
unrar e ~/Downloads/Laika_Beta_1.3.rar
sudo losetup /dev/loop0 ~/ubuntu.img
sudo mount -t ext2 /dev/loop0 ~/ubuntu
sudo rm ~/ubuntu/usr/bin/X11
sudo chmod 777 ~/ubuntu
# Run some commands on your phone
sudo ~/android-sdk-linux_x86/platform-tools/adb shell
mount -o remount,rw -t yaffs2 /dev/block/mtdblock3 /system
mkdir /data/local/mnt
chmod 777 /data/local/mnt
cp /sdcard/bootubuntu /system/bin
chmod 777 /system/bin/bootubuntu
exit
# Push the files to the phone - this takes a LONG time
sudo ~/android-sdk-linux_x86/platform-tools/adb push ~/ubuntu /data/local/mnt
```

Now opening up a terminal on your phone you should be able to run **bootubuntu** as root (you may need to run **su** first). Note that I needed to create a few directories for some things to work:

```
mkdir /var/lib/dpkg/updates
mkdir /etc/apt/preferences.d
mkdir sys
mkdir dev/pts
mkdir proc
sudo apt-get install vim
```

If course, everyone needs to install VIM. EMACS users just don't get it... To start up a GUI, try these commands:

```
export DISPLAY=localhost:0.0
Xvnc -localhost -geometry 854x480 -depth 24 -deferupdate 0
startlxde
```

Now you can run your favorite VNC viewer on localhost:5900 to access a Linux GUI. Pretty cool, eh? Remember, you can't just copy Ubuntu binaries over from your desktop and have them work. You'll need to use things compiled for an ARM processor, most of the things in the Ubuntu repositories (and the Debian repositories from which they're derived) have ARM builds in them so hopefully most of what you need will work just fine.

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>) or follow me on Google+ (<https://profiles.google.com/100268310848930740059>) or Twitter (<http://twitter.com/jdbower>).