

OnePlus 3, plus Nougat

Tl;dr: I hacked at my OnePlus 3 for a few days to update from Marshmallow to Nougat, going the TWRP/SuperSU route. I then swapped stock GApps for OpenGApps pico, to minimize Google's footprint. I ran into several problems, key being the "dm-verity" error message, but it all sorted itself out in the end. This is the whole outline of the battle, but steps 1-7 at the very end as well as their preceding paragraph is basically all I'd need to do in the future.

A [serious camera focusing bug](#) had me shipping my OnePlus 3 out to Toronto for repair (under warranty, completed in record time: it left Monday and was back Wednesday that same week). Still, I was told the repairs could take up to a week so I had to go a whole month before I was able to ship my phone out, and this resulted in many, many [missed photo opportunities](#). The silver lining however, was that I was able to also get them to reset the device to factory settings. Since [minimizing Google's footprint](#) on it earlier this year, the device was working perfectly except I was unable to update from Marshmallow to Nougat on the OS side. At least, not without setting aside a few days to properly read through all the forum talk... and I just didn't have this kind of time, needing my phone running for work much of the week.

The device just came back and I love the new feel of Nougat, but I'm already excited to customize it again and degoogle it as far as possible! Privacy issues are driving me a little bit, of course, but I'm not qualified enough to understand what I'm doing on any real crypto level, and I'm not naïve enough to think I'm doing anything really NSA-proof. Still, doing all this blankets me in something of a feeling of more security, and more importantly, teaches me a lot about how my device works, giving me fuller control over what exactly is going on under the hood. But my main reason for going through with this is I'm just so annoyed at all the auto-updating

bloatware Google serves as part of its default Android experience!

Turns out the whole process took a lot more time and digging (through both forum and phone) to get it done right. Here's the play-by-play.

Day one: First try

First you need to unlock your phone. I did it using the video below:

... but it was very long and had lots of unnecessary talking. But I was patient, it got me through to the end. Here are the main points:

1. Update the system (to Android 7.1.1 currently).
2. [Set up ADB/Fastboot on the PC](#): get ADB Installer v1.4.3 from the Google Drive link on [androidsage](#), run the .exe file and then type "Y" three times, including for getting the additional drivers.
3. Unlock the phone: enable *Developer Options* and then *OEM unlocking*, *advanced reboot*, and *USB debugging*, and then unlock the bootloader. The part where he's fiddling with what appears to be a few buttons @ 5:37 he actually just pushed the power button. At the end, to get to the bootloader screen to confirm the device is unlocked, I had to also re-enable *advanced reboot*.

Next, the device needs to be rooted and then it's good to install a custom recovery program (TWRP has become the standard). I downloaded SR 3 SuperSU v2.79-SR3 from [androidsage](#) and upload it to the phone, but this was a bad idea. You'll see why as you read on. I next downloaded [TWRP 3.1.0](#) from [TWRP](#) into my *adb* folder (C:\adb; [androidsage](#) didn't seem to do this). Shift+right-click in this folder and open command window, then flash the recovery img file into it as per the video. At 6:15 he manually boots into recovery mode,

once the recovery img file has been successfully installed. Here's the whole walk-through:

Now this is important. MAKE A FRESH BACK UP OF EVERYTHING. Check all those boxes you see at 7:30 above. I cannot stress enough how important this is. Lastly install SuperSU. Wipe cache and dalvik; reboot into system. I chose to both prompt to install TWRP and to install it as a system app before rebooting after reading [this article](#) (tl;dr: this allows me to monitor and update TWRP from the app side and not continuously having to do so from the back end). Back up again now that the phone's all set and presto!

All the above is great for now having fuller access over my phone and for being able to make solid backups that I can then export to external drives for safekeeping, but again, the real reason I do it all is to remove bloatware and minimize the amount of apps running on it that may be collecting information on my activity: enter the [Open GApps](#) project. From a [blog post](#) I know OnePlus 3s require the ARM64 platform, and I always get *pico* because that's just enough Google to run the device (any less and it just doesn't work) plus the Play Store to get all my apps like Google Translate and Maps (sadly, still irreplaceable). The steps are simple:

1. Download the appropriate GApps package, then upload that to your phone.
2. Reboot your phone into recovery, then from the TWRP interface select *Install* and navigate to your GApps file, and flash it.

Day two: Second shot



That pesky error.

Everything was working fine until I uploaded the incorrect version of GApps, mistakenly thinking I needed ARM (I needed ARM64). The upload failed so I quickly restored the latest backup above (thank goodness I'd made them this time), and it would install, but then the *Reboot System* button wouldn't show up even 10m after the reboot had clocked 100%. I powered the device down and then up again with the power button and that thankfully rebooted, but I didn't feel secure with this set-up. I tried the previous recovery file and the same thing happened... and then I noticed that upon each restart, I kept getting a confusing [dm-verity](#) error. This may have been because I neglected to properly wipe the cache and data before starting up this process. This step seems important in the forums; otherwise residual footprints from the previous settings seem to mix things up in the new restore. The forums agreed on only one course of action: a full factory reset. It was time for me to learn how to do this. Luckily, it was a cinch:

1. Download the [appropriate OxygenOS version](#) (I always go with the latest *stable* build) into your *adb* folder and then follow the [remaining easy steps](#):
2. Ensure your phone has *OEM unlocking*, *advanced reboot*, and *USB debugging* all enabled. Reboot into recovery mode. From there click through *English > Install from ADB > Upgrade Android from USB? Ok.*
3. Using the ADB sideload method mentioned in the first

part, open a command line in the *adb* folder (Shift+right click or type "cmd" in the folder address bar and hit "Enter"). Verify your device is connected via *adb devices*. You should see a short serial number confirming the device is connected and read by your PC.

4. Type in *adb sideload <filename>*, where "filename" is whatever the name is of the recovery zip file. I renamed my zip to just "oxygen.zip" to make this process easier. Hit Enter and off you go! Here is a video quickly showing how it all looks like in action (I like it because the song is full of hope, and that's what you need at this point in the process):

This took care of that *dm-verity* screen. Next up: redo the first day's work in re-installing TWRP, SuperSU, and then finally the appropriate GApps if you so desire and you should be good to go! Here's [that process](#):

1. Leave the phone on, connect it to your PC, then run the *adb devices* command to ensure it's connected.
2. Enter fastboot mode: *adb reboot bootloader*.
3. Flash TWRP recovery img file (I renamed this to *twrp.img* to make it easier): *fastboot flash recovery twrp.img*.
4. Enter recovery mode, selecting with volume rocker and hitting power to enter.
5. When prompted if you want to keep system "[unmodified and read-only](#)", do that this first time. This is good idea to create a completely fresh backup that you can fall back on [to get OTAs](#) (over-the-air updates: updates that were pushed out after whatever OS version you have). Remember? This was my original problem that had me stuck on Marshmallow months after Nougat came out. Wipe the cache and dalvik (full factory reset is perfect) and then reboot.
6. Enable *Developer Options*, then *OEM unlocking*, *advanced reboot*, and *USB debugging*. If your phone isn't being recognized by the PC, dis- and then re-enable *USB*

debugging. That usually does the trick. Move the TWRP recovery you just made to your PC for safekeeping; move your SuperSU and GApps zip files to your phone. This caused Windows to muck up and restart twice for me so I just unplugged and re-plugged the phone in and it worked fine. Took about 5m though.

7. Reboot back into recovery. For some reason, TWRP did not boot here, I suspect it's because I chose to keep the recovery unmodified and read only in step 5 (this issue is easy to fix; read on). So repeat steps 1-5 to reinstall it. This time it did not offer an option of being read-only so just create a second backup here. When that's done, swipe to a factory reset again under *Wipe* and then swipe right to *Install TWRP App* for the reasons listed earlier. Reboot the device (this will take a minute) and move *that* backup to desktop as well (keep copies of all your backups locally on the device as well). You'll have to go through the welcome screens again. Re-enable *Developer Options*, then *OEM unlocking*, *advanced reboot*, and *USB debugging*.
8. Reboot into recovery once more. This again led me to stock recovery options so I ADB-sideloaded a fresh OxygenOS zip (exactly as before the video in this section). I installed TWRP again, as above. Still, no swipe-right-to-get-out-of-read-only option. At this point I'm too tired to go through figuring out why, and confident enough that I can still de-bloat the phone, at least sufficiently to meet my needs.
9. In the TWRP recovery screen I *Wipe* and do a factory reset. I *Install* my SuperSU, then *wipe cache/dalvik*. Then GApps is installed. This is where I get nervous because I *really* don't want to go through any more of this... but it's looking good. Done! *Wipe cache/dalvik* again, reboot system with the TWRP installation prompt enabled and there we have it! A de-bloated, largely de-googled OnePlus 3 running the latest Android OS: Nougat!!!

... and all was good until I realized the SIM card wasn't being picked up and I had no signal. That just won't do. TWRP was very confusing because of that initial mount setting (it wasn't letting me upload backups because of being in read-only mode), but that's taken care of through [unchecking a box](#) under *Mount*. So off to the official site for the official [recovery img](#) file, sideload that in place of TWRP, and then re-load a fresh OxygenOS ROM back onto the phone using it. At this point I just don't have more time to dabble in all this, and it's a pity too – the minimized Google footprint felt great. I'm leaving this phone with how it was when I received it, and hoping the forums figure out how to not kill the SIM signal with GApps pico in future releases. Fingers crossed!

The main lesson today was how to backup and factory-reset my device on a deeper level. This feels like finding those one-up extra life mushrooms in Super Mario way back when, and is very good for selling it later without any sensitive information on it, from documents to passwords to bitcoin (yikes).

Insert 30m and some chewy Chinese candy...

Done! With a signal! Turns out my SuperSU file was faulty (either *BETA-SuperSU-v2.74-2-20160519174328-forced-systemless.zip* or *SR3-SuperSU-v2.79-SR3-20170114223742.zip*, both from the independent site androidsage). But XDA-developers pulled through again: they have a whole forum conversation dedicated to [getting SuperSU right](#) on there. I went with [SuperSU-v2.79-201612051815](#) and it works like a charm! Still getting the *dm-verity* error but at this point this is a very minor problem (turns out it's [not a big deal](#), just a system error message, kind of like a “check engine” light usually is on old cars). Okay, time for a wrap-up. Let's see if I can't get the device up to factory settings, then make a TWRP backup, then bring it up to Google Minimized status in under 15m. Here is the setup:

First a TWRP backup of the current state to the PC. SuperSU

and GApps pico zips are already on the device, ADB is running on the PC, *Developer Options* plus *OEM unlocking*, *advanced reboot*, and *USB debugging* are all enabled, and the device is connected and charged at 100%. Files: [ADB](#) from [AndroidSage](#), [OnePlus OxygenOS 4.1.3 \(stock & recovery\)](#), [SuperSU-v2.79-201612051815](#) from [XDA-developers](#), and [OpenGApps ARM64 7.1 pico](#). Now the process:

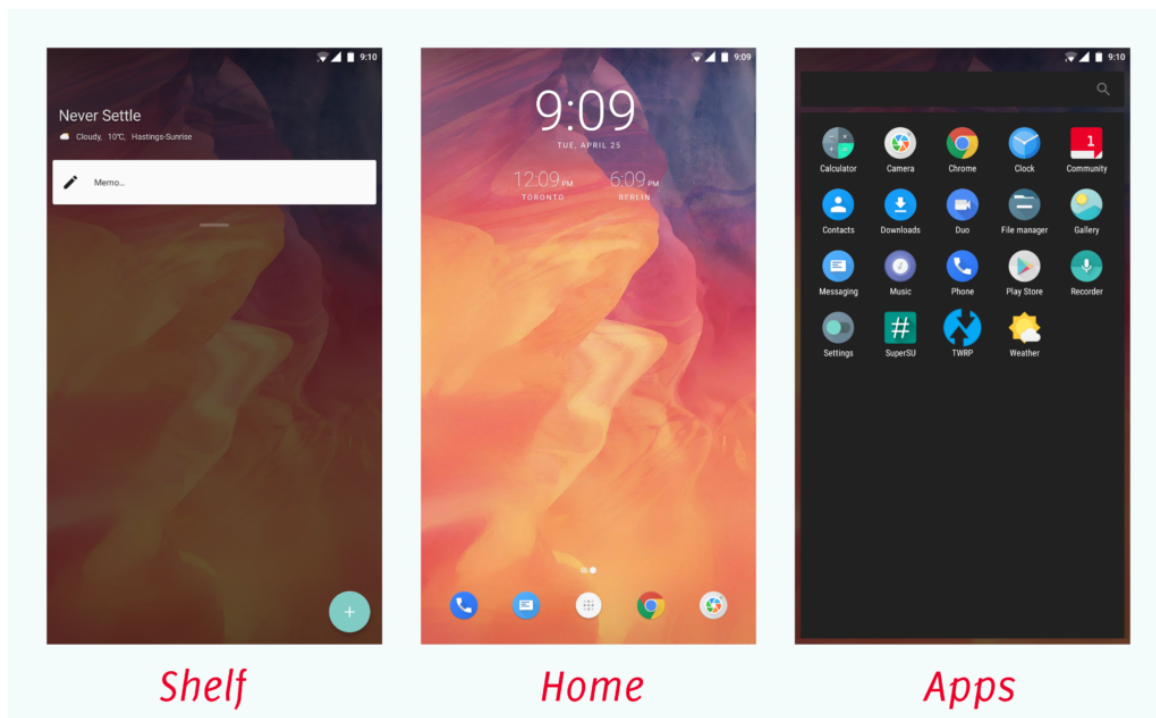
1. PC (ADB flash stock recovery): *adb devices > adb reboot bootloader > fastboot flash recovery <filename>.img*. Reboot into recovery with volume rockers.
2. OnePlus: *English > Install from ADB > Upgrade Android from USB? Ok*.
3. PC (ADB sideload stock OS): *verify via adb devices > adb sideload <filename>.img*. Reboot into system.
4. OnePlus: *Reboot. Settings > Developer Options > Dis-* then reen-able *USB Debugging*.
5. PC (ADB flash TWRP recovery): *verify via adb devices > adb reboot bootloader > fastboot flash recovery <filename>.img*. Reboot into recovery with volume rockers.
6. OnePlus (flash SuperSU): *Install > SuperSU<filename>.zip > Wipe cache/dalvik > Backup > Check Boot, System, Data > Swipe to Backup*.
7. OnePlus (flash OpenGApps): *Install > OpenGApps<filename>.zip > Wipe cache/dalvik > Backup > Check Boot, System, Data > Swipe to Backup. Reboot system > Swipe to Install TWRP App*.

Time-wise, each step above took under a minute apart from sideloading the stock OS (15m). Altogether, the entire process lasted 25m. And the best part? No *dm-verity* error messages!

Day Three: After the Smoke Cleared

Waking up bright and early, I already couldn't wait to get my hands on my machine again and start doing the *fun* stuff: uploading all my apps and setting up the settings to be as

clean and clear and soft and solid as possible. OnePlus' shelf feature is amazing here (I no longer need extra note-taking or weather apps)! Here's how she looks now:



What a beauty! Look at that: fresh TWRP, SuperSU, and OpenGApps pico in the back end, and then personalized for productivity up front. And all this right through stock options!