

xda

NEWSFORUMSDEVICESTOPICSBEST POSTS

#Pixel 4#Galaxy S20#5G#OnePlus Nord#Huawei#Snapdragon#Xiaomi#Huawei HMS#Samsung#ROG Phone 3#Huawei Developers#POCO#Re

xda-developers > Android Development and Hacking > Android Software and Hacking General [Developers Only] > [REFERENCE] How to compile an Android kernel by nathanchance

THE LANGSTON
EPPING

“100M FROM EPPING TRAIN STATION,
WITH LINKS TO EVERYWHERE.”


Richard K, The Langston Purchaser

LIFESTYLE RESIDENCES
ABOVE A BRAND
NEW RETAIL PRECINCT

ENQUIRE NOW

cbus
property

CBRE



[REFERENCE] How to compile an Android kernel

By [nathanchance](#), Senior Recognized Developer / Recognized Contributor on 26th June 2017, 04:38 AM

13,748 posts

Thanks: 50,864

Tweet

Like

FORUMS

Android Software and Hacking General [Developers Only]

Android General

Android Q&A, Help & Troubleshooting

Huawei Developers

Android Software Development

[More]

1231151Last >>

Search Thread

Introduction

Hello everyone, I will be going over how to compile a kernel from beginning to end!

Prerequisites:

- A Linux environment (preferably 64-bit)
- Knowledge of how to navigate the command line
- Common sense
- A learning spirit, there will be no spoonfeeding here

What this guide will cover:

- Downloading the source
- Downloading a cross compiler
- Building the kernel
- Flashing the kernel

What this guide will NOT cover:

- Setting up a build environment (plenty of existing Linux installation guides)
- Adding features to the kernel (plenty of git cherry-picking guides)

I know this has been done before but on a cursory search, I have not seen a guide that was recently updated at all.

1. Downloading the source

CLICK TO HIDE CONTENT

If you have a custom kernel you want to build, move along after cloning the kernel using the git clone command below.

If you are compiling your stock kernel, it is ultimately up to you to know where to get your kernel source from but here are some common places:

Google: <https://android.goesource.com/kernel/msm/> (pick your architecture and look at the branches)

LG: <http://opensource.lge.com/index>

Samsung: <http://opensource.samsung.com/reception.do>

HTC: <https://www.htcdev.com/devcenter/downloads>

OnePlus: <https://github.com/OnePlusOSS>

Motorola: <https://github.com/MotorolaMobilityLLC>

Sony: <https://github.com/sonyxperiadev/kernel>

MOST THANKED

ALL-TIME

RECENT

19

Compile Android Pie kernels for the Pixel (XL) and Pixel 2 (XL) Google ...

nathanchance OP

2018-Aug-07 16:46

8

(quote) Unfortunately, my time nowadays is limited between my ...

nathanchance OP

2020-Mar-02 12:57

https://forum.xda-developers.com/android/software-hacking/reference-how-to-compile-android-kernel-t3627297

1/9

To download the kernel, you can either use git clone or download the tarball and extract it:

Code:

```
git clone -b <branch_to_checkout> <url> <desired_folder>

OR
```

For example, if I wanted to grab the latest Nexus 6P from Google above:

Code:

```
git clone -b android-msm-angler-3.10-nougat-mr2 https://...
```

This will clone the kernel/msm repo into an angler folder and checkout the android-msm-angler-3.10-nougat-mr2 automatically.

I can try and help you locate your source if necessary.

2. Downloading a cross compiler

CLICK TO HIDE CONTENT

Since most Android devices are ARM based, we need a compiler that is targeting ARM devices. A host (or native) compiler will not work unless you are compiling on another ARM device.

You can either compile one yourself if you know how (crosstool-NG is a great tool for this) or download a prebuilt one. Luckily Google provides a high quality toolchain for this, in both an [arm \(32-bit\)](#) and [arm64 \(64-bit\)](#). It's up to you to know the architecture of your device. Typically speaking, most devices in the past two-three years are 64-bit.

Another popular toolchain is UberTC, which can be found here: <https://bitbucket.org/matthewdalex/>. Most kernels will need patches for anything higher than 4.9 and while I don't mind assisting with finding them, you should compile with Google's toolchain first.

Once you have decided, clone the toolchain:

Code:

```
git clone <url>
```

3. Compile the kernel

CLICK TO HIDE CONTENT

1. Point the Makefile to your compiler (run this from within the toolchain folder!!)

Code:

```
export CROSS_COMPILE=$(pwd)/bin/<toolchain_prefix>-
```

Example:

Code:

```
export CROSS_COMPILE=$(pwd)/bin/aarch64-linux-android-
```

NOTE #1: For kernels that can be compiled with Clang (like the Pixel 2), see [this guide](#). I will support it here if there are any questions.

NOTE #2: Pixel and Pixel 2 users, you will need to follow [these steps](#) as well if compiling for Android Pie.

2. Tell the Makefile the architecture of the device

Code:

```
export ARCH=<arch> && export SUBARCH=<arch>
```

- 7 (quote) Just skip the make clean for now and go straight to ... 2018-Aug-22 14:45
- 7 (quote) On a scale of 1-10 how much Off-Topic is allowed ? ... 2017-Jun-29 12:33
- 6 You're on fire with this kernel stuff:D Sent from my LEX727 ... 2017-Jun-25 23:39

THREAD SEARCH

Search this thread

SUGGESTED APPS



Official XDA Forum App
The XDA App is the fastest way to access the forums on mobile.



Navigation Gestures
Customizable gesture control for any Android device.



XDA Labs
Labs is an independent app store that gives developers full control over their work.



Substratum
The ultimate, most complete theming solution for Android.



XDA Feed
The best way to get cutting edge news about your device!

TOP FORUM DISCUSSIONS

Here's how to unlock the bootloader of the Verizon Google Pixel/Pixel XL running Android 10
🕒 July 17, 2020

The OnePlus 8 Pro's color filter camera is still accessible with an ADB command
🕒 July 17, 2020

HTC Desire 20 Pro can be bootloader unlocked with a simple Fastboot command
🕒 July 14, 2020

Fix the Razer Phone 2's camera viewfinder and EIS with this Magisk Module
🕒 July 11, 2020

Descendant X custom ROM for the Xiaomi Mi A1 and Mi 9 adds "Guardia," a background permission monitor
🕒 July 11, 2020

Unleash the true performance of the Red Magic 5G with this custom kernel
🕒 July 9, 2020

Example:

Code:

```
export ARCH=arm64 && export SUBARCH=arm64
```

3. Locate your proper defconfig

Navigate to the arch/<arch>/configs folder within the kernel source (e.g. arch/arm64/configs) and locate your device's or custom kernel developer's proper config file. For example, it will often be in the form of <codename>_defconfig or <kernel_name>_defconfig. Generic Qualcomm configs may be used as well (msm-perf_defconfig, msmcortex-perf_defconfig). When in doubt, ask here if you are confused. A defconfig tells the compiler what options to add to the kernel.

4. Build the kernel

Code:

```
make clean
make mrproper
make <defconfig_name>
make -j$(nproc --all)
```

If those commands succeed, you will have an Image, Image-dtb, Image.gz, or Image.gz-dtb file at the end.

If it failed, as was pointed out to me by @flar2 while making a complete idiot of myself, you may need to specify an output directory while making new CAF based kernels, like so:

Code:

```
mkdir -p out
make O=out clean
make O=out mrproper
make O=out <defconfig_name>
make O=out -i$(nproc --all)
```

If after that something is still broken, you may need to fix some headers or other issues. If it is a custom kernel, bring it up with your developer.

If it's an OEM, it's up to you to try and fix it, which we can assist with.

4. Flash the kernel

CLICK TO HIDE CONTENT

Assuming you were able to compile the kernel successfully, you now need to flash it! I will be covering two different ways to flash a compiled kernel: unpacking and repacking the boot image by hand using Android Image Kitchen or AnyKernel2, both by the brilliant @osm0sis. If there are any per-device nuances, please let me know and I'll add them here! Additionally, this section can vary drastically by device, you may need to consult developers of your device for assistance if necessary.

Android Image Kitchen

CLICK TO HIDE CONTENT

1. Pull your device's boot image from the latest image available for your device (whether it be a ROM or stock)
2. Download the latest Android Image Kitchen from [this thread](#).
3. Run the following with the boot image:

Code:

```
unpackimg.sh <image_name>.img
```

4. Locate the zImage file and replace it with your kernel image (rename it to what came out of the boot image)
5. Run the following to repack:

Code:

```
repackimg.sh
```

6. Flash the new boot image with fastboot or TWRP!

How to restore Widevine L1 DRM on the OPPO Find X2 Pro to stream Netflix and Amazon Prime Video in HD

July 7, 2020

AnyKernel2

CLICK TO HIDE CONTENT

1. Download the latest AnyKernel2 zip: <https://github.com/osm0sis/AnyKernel2/master.zip>
2. Apply [this patch](#) to clean out all of the demo files:

Code:

```
wget https://github.com/nathanchance/AnyKernel2/commit/add
patch -p1 < addb6ea860aab14f0ef684f6956d17418f95f29a.diff
rm addb6ea860aab14f0ef684f6956d17418f95f29a.diff
```

3. Place your kernel image in the root of the file.
4. Open the anykernel.sh file and modify the following values:
 - kernel.string: your kernel name
 - device.name#: List all of your device's codenames (from the /system/build.prop: ro.product.device, ro.build.product)
 - block: Your boot image's path in your fstab. The fstab can be opened from the root of your device and it will look something like this:
https://android.googlesource.com/dev_r/fstab.angler
The first column is the value you want to set block to.
5. After that, zip up the kernel and flash it!

Code:

```
zip -r9 kernel.zip * -x README.md kernel.zip
```

Tips and tricks

CLICK TO HIDE CONTENT

1. Remove GCC wrapper

CLICK TO HIDE CONTENT

A lot of kernels from CAF include a Python script that will essentially turn on -Werror, causing your build to error at the most benign stuff. This is necessary with higher GCC versions as there are a lot more warnings.

Here is the diff of what you need to change in the Makefile:

Code:

```
diff --git a/Makefile b/Makefile
index laaa760f255f..bfccd5594630 100644
--- a/Makefile
+++ b/Makefile
@@ -326,7 +326,7 @@ include $(srctree)/scripts/Kbuild.in

AS      = $(CROSS_COMPILE)as
LD      = $(CROSS_COMPILE)ld
-REAL_CC = $(CROSS_COMPILE)gcc
+CC      = $(CROSS_COMPILE)gcc
CPP      = $(CC) -E
AR      = $(CROSS_COMPILE)ar
NM      = $(CROSS_COMPILE)nm
@@ -340,10 +340,6 @@ DEPMOD  = /sbin/depmod
PERL     = perl
CHECK    = sparse

-# Use the wrapper for the compiler. This wrapper scans
```

2. Using a higher level GCC toolchain

CLICK TO HIDE CONTENT

Using a higher GCC toolchain (5.x, 6.x, 7.x or even 8.x) will require you to nuke the GCC wrapper script as

above and use a unified GCC header file (pick the following if you have an include/linux/compiler-gcc#.h file):

3.4/3.10: <https://git.kernel.org/pub/scm/linux...9bb8868d562a8a>
3.18: <https://git.kernel.org/pub/scm/linux...9f67d656b1ec2f>

You may get a lot of warnings but they are not entirely necessary to fix.

3. Adding upstream Linux to kernel source

CLICK TO HIDE CONTENT

Once you have gotten familiar with git and the compilation process, you should consider [upstreaming your kernel](#). This will allow you to stay on top of CVE and bug fixes by staying up to date with the latest work of the Linux kernel developers.

Receiving help

I am happy to answer anything that I touched on in this guide. I may point you to another thread if it's better suited but I don't mind off topic (within reason) within the thread. I also want this to be a collaborative effort; other developers, if you have something to add, correct, or improve upon, please let me know!

I am particular in how people ask for help. I do NOT respond to posts asking for a hand out ("How do I fix this?", "Please fix this!", etc.). I only respond to posts with clear logs and steps that you have tried. Basically, show me that you have read this guide and have a specific issue. I am not here to hold your hand through this, this is a developers' forum.



The Following 203 Users Say Thank You to nathanchance For This Useful Post: [View]



食品科学及安全（荣誉）
理学士



[mosimchah](#)
Senior Member

Thanks: 9,143



26th June 2017, 04:39 AM | #2

You're on fire with this kernel stuff 🙌

Sent from my LEX727 using XDA Labs



The Following 6 Users Say Thank You to mosimchah For This Useful Post: [View] Gift mosimchah Ad-Free

[NoNameMi](#)
Junior Member

Thanks: 8



26th June 2017, 05:15 AM | #3

Nice and clear introduction on how to build a kernel. This small how-to was so simple, I can keep it in my head! Awesome!

The Following 2 Users Say Thank You to NoNameMi For This Useful Post: [View]

Gift NoNameMi Ad-Free

Subham

Senior Member

Thanks: 2,062

26th June 2017, 05:20 AM | #4

A really helpful guide much needed around for upcoming developers. This is the perfect guide for them. Nice work 😊

Sent from my ONE A2003 using Tapatalk

The Following User Says Thank You to Subham For This Useful Post: [View]

Gift Subham Ad-Free

NZnewbie

Senior Member

Thanks: 213

26th June 2017, 09:38 AM | #5

The guys a Feckn Champ.

Thanks a lot!!.

The Following 3 Users Say Thank You to NZnewbie For This Useful Post: [View]

Gift NZnewbie Ad-Free

jacbarahona72

Senior Member

Thanks: 209

27th June 2017, 08:13 PM | #6

Thanks @The Flash

Enviado desde mi Nexus 6 mediante Tapatalk

The Following 2 Users Say Thank You to jacbarahona72 For This Useful Post: [View]

Gift jacbarahona72 Ad-Free

karandpr

Senior Moderator / Developer Commit...

29th June 2017, 05:33 PM | #7

Thanks: 13,086

Quote:
Originally Posted by The Flash

Introduction

I am happy to answer anything that I touched on in this guide. I may point you to another thread if it's better suited but I don't mind off topic (within reason) within the thread. I also want this to be a collaborative effort; other developers, if you have something to add, correct, or improve upon, please let me know!

I am particular in how people ask for help. I do NOT respond to posts asking for a hand out ("How do I fix this?", "Please fix this!", etc.). I only respond to posts with clear logs and steps that you have tried. Basically, show me that you have read this guide and have a specific issue. I am not here to hold your hand through this, this is a developers' forum.

On a scale of 1-10 how much Off-Topic is allowed ?

CLICK TO HIDE CONTENT

CLICK TO HIDE CONTENT

CLICK TO HIDE CONTENT

CLICK TO HIDE CONTENT

Nice guide ..

The Following 7 Users Say Thank You to karandpr For This Useful Post: [View]

Gift karandpr Ad-Free

kevintm78

Senior Member

29th June 2017, 07:17 PM | #8

Thanks: 2,469

I have been using ./build_kernel.sh to compile kernels as was suggested by another guide and I'm wondering if there's any pros or cons doing it that way as opposed to using the make defconfig way.

They seem to be working ok but this is the second guide on xda that suggest the way you're doing it and I'm definitely open to change if this way is better. Any thoughts on the two methods would be much appreciated. I also would like to say thanks for these new guides as finding kernel dev info for newbies is very scarce and mostly outdated. I really look forward to seeing this thread take off.

The Following 2 Users Say Thank You to kevintm78 For This Useful Post: [View]

Gift kevintm78 Ad-Free

nathanchance

OP Senior Recognized Developer / Re...

29th June 2017, 07:39 PM | #9

Thanks: 50,864

https://forum.xda-developers.com/android/software-hacking/reference-how-to-compile-android-kernel-t3627297

7/9

Quote:
Originally Posted by [kevintm78](#)

I have been using ./build_kernel.sh to compile kernels as was suggested by another guide and I'm wondering if there's any pros or cons doing it that way as opposed to using the make defconfig way.


They seem to be working ok but this is the second guide on xda that suggest the way you're doing it and I'm definitely open to change if this way is better. Any thoughts on the two methods would be much appreciated. I also would like to say thanks for these new guides as finding kernel dev info for newbies is very scarce and mostly outdated. I really look forward to seeing this thread take off. 🙌

Well this build_kernel.sh script is most likely doing just this. I personally use a script to compile and package my kernel, I would never do the "manual" way like this once you know how.

↩

🔗

The Following 3 Users Say Thank You to [nathanchance](#) For This Useful Post: [\[View \]](#)

[nathanchance](#)

OP Senior Recognized Developer / Re...

Thanks: 50,864

29th June 2017, 08:01 PM | #10

🐦

f


I have updated the OP with a note about compiling newer CAF releases (3.18 and 4.4 to my knowledge). As was pointed out by [@flar2](#) while I was making an idiot of myself accusing him of violating the GPL (for which I truly do apologize), you may need to specify an output directory (by adding an O= flag). This is actually done automatically when a ROM compiles a kernel inline so you will only run into this while compiling standalone.

I have added it to my script [here](#) if you want an idea of how to add it to yours.

↩

🔗

The Following 4 Users Say Thank You to [nathanchance](#) For This Useful Post: [\[View \]](#)

[AhmAdDev99](#)

Member

Thanks: 21

5th July 2017, 04:26 PM | #11


🐦

f

So here i'am what should i do to fix the initramfs problem?
I tried "chmod -R a+x kernel" but i still get the same problem.

↩

🔗



华为云


0元试用

40+高配云产品

云主机 | 安全 | 数据库

企业0元体验

立即体验



广告 X

↩ Reply

🔔 Subscribe to Thread

Page 1 of 76

1

2

3

11

51

➤

Last ➤➤

Guest Quick Reply (no urls or BBcode)

https://forum.xda-developers.com/android/software-hacking/reference-how-to-compile-android-kernel-t3627297

8/9

Guest Quick Reply (no urls or BBcode)

Message:

Post Quick Reply

Go Advanced

Previous Thread

Next Thread

Top Threads in Android Software and Hacking General [Developers Only] by ThreadRank

Starwars edition themes icon pack for third party launcher like nova

Android Apps and Games

22nd December 2019

[TOOL] Advance Tool for Android™ - ADB and fastboot commands

Android Software and Hacking General [Developers Only]

21st October 2019

[Android_x86] PhoenixOS DarkMatter - Supercharged for Gamers!

Miscellaneous Android Development

4th November 2019

[GUIDE] Rooting the UMIDIGI F2 | Magisk | TWRP

Android General

5th January 2020

[APK] Coronavirus Tracker for Android - Corona Smart Tracker

Android Apps and Games

5th March 2020

[TOOLS][ZIPS][SCRIPTS]Android Backup and Restore Tools[Multiple Devices/Platforms]

Android Software and Hacking General [Developers Only]

8th December 2019

Bliss OS (x86) for PC's - 12.x Development - Beta Builds

Android Software Development

16th November 2019

[APP] XiaoAi Lite Multi Language (English for now)

Android Apps and Games

28th December 2019

Does Your Life Fall Flat ?
Chance to win a HUAWEI Matebook 13 or Watch GT2
FIND OUT!

Home

xda-developers

Android Development and Hacking

Android Software and Hacking General [Developers Only]

[REFERENCE] How to compile an Android kernel by nathanchance

xdadevelopers

XDA Developers was founded by developers, for developers. It is now a valuable resource for people who want to make the most of their mobile devices, from customizing the look and feel to adding new functionality.
Are you a developer? | Terms of Service

We're Social

Twitter

Discord

Facebook

YouTube

Telegram

Instagram

Hosted by

leaseweb

mobile hosting

Thei

Technological and Higher Education Institute of Hong Kong
香港高等教育科技學院

食品科学及安全（荣誉）
理学士

广告

×

VTC

机构成员

-- XDA 2015

More info

Contact

Advertise

Careers

Rules

Suggest Content

Security

Privacy Policy

XDA App

Remove ads on XDA

Useful Links

Redmi Note 8 Pro

Software Update Download Links

Root Any Device

How To Guides

XDA's Best

Copyright © xda-developers. Hosted by Leaseweb

https://forum.xda-developers.com/android/software-hacking/reference-how-to-compile-android-kernel-t3627297

9/9