Contents

[Sources 2](#_Toc75359973)

[Manual: 2](#_Toc75359974)

[Issues: 2](#_Toc75359975)

[Manually building signal-cli v0.8.3 for on a Raspberry Pi 2](#_Toc75359976)

[Install required packages to be able to build signal-cli and dependent zklibrary + libsignal-client libraries 2](#_Toc75359977)

[Building signal-cli 2](#_Toc75359978)

[Setting the video memory to 16MB: 3](#_Toc75359979)

[Getting the free memory: 3](#_Toc75359980)

[Get the signal-cli from source: 3](#_Toc75359981)

[Build the client: 3](#_Toc75359982)

[Get and update the libraries: 3](#_Toc75359983)

[Get and build zkgroup library: 3](#_Toc75359984)

[Get and build libsignal-client library: 4](#_Toc75359985)

[Update jar file with updated signal-client-java-\*.so file 4](#_Toc75359986)

[Update jar file with updated libzkgroup.so file 4](#_Toc75359987)

[Copy the updated JAR Files (for use with tar and zip) 4](#_Toc75359988)

[Update ZIP file 5](#_Toc75359989)

[Update TAR file 5](#_Toc75359990)

[Finished signal-cli-0.8.3.tar file: 5](#_Toc75359991)

[Install/export signal-cli 5](#_Toc75359992)

[Unregister previous version 6](#_Toc75359993)

[Registering 6](#_Toc75359994)

[Sending a test message: 6](#_Toc75359995)

[Install required packages: 7](#_Toc75359996)

[Update the current packages / raspbian: 7](#_Toc75359997)

[Install latest JAVA JDK (v11.0.11+9-post-Raspbian-1deb10u1) 7](#_Toc75359998)

[Install version v7.1 of Gradle 7](#_Toc75359999)

[Install Rust: 8](#_Toc75360000)

[Install protobuf-compiler: 8](#_Toc75360001)

[Install Git 8](#_Toc75360002)

[Install Git with apt-get (v2.20.1) 8](#_Toc75360003)

[Install latest stable version of Git from source (v2.32.0) 8](#_Toc75360004)

[Install zip and unzip 9](#_Toc75360005)

[Reboot 9](#_Toc75360006)

Sources:

Read/see these source pages first, for source code/author of signal-cli:

<https://github.com/AsamK/signal-cli>

<https://github.com/AsamK/signal-cli/wiki/Provide-native-lib-for-libsignal>

# Manual:

<https://github.com/AsamK/signal-cli/blob/master/man/signal-cli.1.adoc>

# Issues:

<https://github.com/AsamK/signal-cli/discussions/393#discussioncomment-246169>

https://github.com/AsamK/signal-cli/issues/643

# Manually building signal-cli v0.8.3 for on a Raspberry Pi

Below are the steps which worked for me, in June 2021, on a **Raspberry Pi 3B with Raspbian GNU/Linux 10 (buster)**

* On a clean Raspbian OS install, the directory “~” is the home directory of the current user. That is “/home/pi” as default.
* User feedback shows this also works on a Raspberry Pi 4, but as I don’t have one, have not tested that myself.
* Reference: <https://github.com/AsamK/signal-cli/wiki/Provide-native-lib-for-libsignal>

## Install required packages to be able to build signal-cli and dependent zklibrary + libsignal-client libraries

- Raspberry PI 3B with Raspbian GNU/Linux 10 (buster)

<https://www.raspberrypi.org/software/operating-systems/#raspberry-pi-os-32-bit>

Raspbian buster Lite image, was downloaded on 23-Jun-2021 + full upgrade done using: sudo apt-get update && apt-get full-upgrade

Checking the kernel version: **uname -a**

Linux raspberrypi 5.10.17-v7+ #1421 SMP Thu May 27 13:59:01 BST 2021 armv7l GNU/Linux

- Java JDK v11.0.11+9-post-Raspbian-1deb10u1

- Gradle v7.1

- Rust

rustup 1.24.3 (ce5817a94 2021-05-31)

rustc 1.53.0 (53cb7b09b 2021-06-17)

- protobuf-compiler 3.6.1.3-2+rpi1

- git/stable,now 1:2.20.1-2+deb10u3

- zip and unzip

If you don’t have these installed, see the section “*Install required packages*:”

## Building signal-cli

Building signal-cli on a raspberry pi 3 is not easy due to its limited memory. It needs a lot of free memory!

You’ll need to (temporarily) stop/kill any unneeded processes/services and reduce the Video memory to 16MB

The raspberry pi needs a *lot* of free memory to build signal-cli

For v0.8.3, 850MB of available memory (as seen with free -m) was needed

With 807MB free it failed to build. Also used the --stacktrace option for the gradle build.

### Setting the video memory to 16MB:

$ **sudo raspi-config**

* **4 Performance Options**
* **P2 GPU Memory**
* Type in the value: **16**
* **<Ok>**
* **<Finish>**
* Reboot?
* **<Yes>**

### Getting the free memory:

This is the free memory I got on my pi 3, after installing all the required apps + reboot:

**free -m**

total used free shared buff/cache available

Mem: 973 35 864 6 73 882

Swap: 99 0 99

### Get the signal-cli from source:

<https://github.com/AsamK/signal-cli> "Building" section:

$ **mkdir -p ~/\_downloaded**  
$ **cd ~/\_downloaded**

if you don’t have the source yet:  
$ **git clone**[**https://github.com/AsamK/signal-cli.git**](https://github.com/AsamK/signal-cli.git)

$ **cd signal-cli**

else update the local repository:

$ **git pull**

Make a local branch out of the release/tag v0.8.3:

$ **git checkout -b br\_v0.8.3 v0.8.3**

### Build the client:

$ **gradle build --stacktrace** (note this takes several minutes (~16min) to complete)

Note: if the build fails and you want to clean-up the build files and old ‘gradle deamons’, to try again a clean build, remove the “~/\_downloaded/signal-cli/.gradle” folder and content. **rm -rf ~/\_downloaded/signal-cli/.gradle**

$ **gradle installDist**  
$ **gradle distTar**

**The 2 files are now:**

~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.tar

~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.zip

Both will still need to have the 2 libraries updated.

## Get and update the libraries:

### Get and build zkgroup library:

Check the version of the zkgroup-java-x.x.x.jar file in ~/\_downloaded/signal-cli/build/install/signal-cli/lib:

$ **ls -hal ~/\_downloaded/signal-cli/build/install/signal-cli/lib/zkgroup-java-\***

-rw-r--r-- 1 pi pi 607K Jun 20 13:53 /home/pi/\_downloaded/signal-cli/build/install/signal-cli/lib/zkgroup-java-0.7.0.jar

For signal-cli version v0.8.3, this is still ‘zkgroup-java-0.7.0.jar’

We will be updating using v0.7.2 for this:

Download the Source code for zkgroup library v0.7.2 (<https://github.com/signalapp/zkgroup/releases>)  
$ **wget -O ~/\_downloaded/zkgroup-0.7.2.zip https://github.com/signalapp/zkgroup/archive/v0.7.2.zip**

Extract content of zkgroup-0.7.2.zip to ~/\_downloaded (it will create subfolder /zkgroup-0.7.2):  
$ **unzip -d ~/\_downloaded ~/\_downloaded/zkgroup-0.7.2.zip**

$ **cd ~/\_downloaded/zkgroup-0.7.2**  
$ **make libzkgroup** (note this takes several minutes (~21min) to complete)

Once finished, the required file is: ~/\_downloaded/zkgroup-0.7.2/target/release/libzkgroup.so

### Get and build libsignal-client library:

Clone the GIT source:

$ **cd ~/\_downloaded**

$ **git clone https://github.com/signalapp/libsignal-client.git**

Check the version of the signal-client-java-x.x.x.jar file in ~/\_downloaded/signal-cli/build/install/signal-cli/lib:

$ **ls -hal ~/\_downloaded/signal-cli/build/install/signal-cli/lib/signal-client-java-\***

-rw-r--r-- 1 pi pi 2.0M Jun 20 13:52 /home/pi/\_downloaded/signal-cli/build/install/signal-cli/lib/signal-client-java-0.2.3.jar

For signal-cli version v0.8.3, this is ‘signal-client-java-0.2.3.jar’

$ **cd ~/\_downloaded/libsignal-client**

List the existing tags: **git tag -l**

Create local branch from tag/release ‘java-0.2.3’:

$ **git checkout -b br\_java-0.2.3 java-0.2.3**

Build the Library:

$ **cd java**

# Prevent building the android library

$ **sed -i "s/, ':android'//" settings.gradle**

$ **./build\_jni.sh desktop**

The native lib can now be found in the libsignal-client directory:

~/\_downloaded/libsignal-client/target/release/libsignal\_jni.so

### Update jar file with updated signal-client-java-\*.so file

$ **zip -ur ~/\_downloaded/signal-cli/build/install/signal-cli/lib/signal-client-java-0.2.3.jar ~/\_downloaded/libsignal-client/target/release/libsignal\_jni.so**

adding: home/pi/\_downloaded/libsignal-client/target/release/libsignal\_jni.so (deflated 55%)

### Update jar file with updated libzkgroup.so file

$ **zip -ur ~/\_downloaded/signal-cli/build/install/signal-cli/lib/zkgroup-java-0.7.0.jar ~/\_downloaded/zkgroup-0.7.2/target/release/libzkgroup.so**

adding: home/pi/\_downloaded/zkgroup-0.7.2/target/release/libzkgroup.so (deflated 55%)

### Copy the updated JAR Files (for use with tar and zip)

$ **cd ~/\_downloaded/signal-cli/build/distributions**  
$ **mkdir -p signal-cli-0.8.3/lib/**

Copy both updated libraries:  
$ **cp ~/\_downloaded/signal-cli/build/install/signal-cli/lib/zkgroup-java-0.7.0.jar signal-cli-0.8.3/lib/**

$ **cp ~/\_downloaded/signal-cli/build/install/signal-cli/lib/signal-client-java-0.2.3.jar signal-cli-0.8.3/lib/**

Check that the 2 files are there:

$ **ls -hal signal-cli-0.8.3/lib/**

### Update ZIP file

$ **zip -ur ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.zip signal-cli-0.8.3/lib/zkgroup-java-0.7.0.jar**

updating: signal-cli-0.8.3/lib/zkgroup-java-0.7.0.jar (deflated 1%)

$ **zip -ur ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.zip signal-cli-0.8.3/lib/signal-client-java-0.2.3.jar**

updating: signal-cli-0.8.3/lib/signal-client-java-0.2.3.jar (deflated 1%)

### Update TAR file

$ **tar --delete -vPf ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.tar signal-cli-0.8.3/lib/zkgroup-java-0.7.0.jar**  
$ **tar --owner='' --group='' -rvPf ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.tar signal-cli-0.8.3/lib/zkgroup-java-0.7.0.jar**

signal-cli-0.8.3/lib/zkgroup-java-0.7.0.jar

$ **tar --delete -vPf ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.tar signal-cli-0.8.3/lib/signal-client-java-0.2.3.jar**  
$ **tar --owner='' --group='' -rvPf ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.tar signal-cli-0.8.3/lib/signal-client-java-0.2.3.jar**

signal-cli-0.8.3/lib/signal-client-java-0.2.3.jar

## Finished signal-cli-0.8.3.tar file:

Updated signal-cli-0.8.3.tar is now in: ~/\_downloaded/signal-cli/build/distributions/

## Install/export signal-cli

$ **sudo tar xf ~/\_downloaded/signal-cli/build/distributions/signal-cli-0.8.3.tar -C /opt**  
$ **sudo ln -sf /opt/signal-cli-0.8.3/bin/signal-cli /usr/local/bin/**

* signal-cli is now installed on the Raspberry Pi
* You can check the now installed version with:
  + $ **signal-cli --version**

signal-cli 0.8.3

# Unregister previous version

See the online signal-cli manual. Below is what I used from it:

/opt/signal-cli-0.8.1/bin/signal-cli -u <full\_international\_mobile\_number> unregister

# Registering

See the online signal-cli manual. Below is what I used from it:

signal-cli -u <full\_international\_mobile\_number> register

if getting an error about captcha:

open browser

F12 and then Console

***--captcha***

The captcha token, required if registration failed with a captcha required error. To get the token, go to <https://signalcaptchas.org/registration/generate.html> Check the developer tools (F12, console) for a redirect starting with signalcaptcha://

Everything after signalcaptcha:// is the captcha token.

signal-cli -u <full\_international\_mobile\_number> register --captcha <captcha token>

signal-cli -u <full\_international\_mobile\_number> verify <CODE>

***--config CONFIG***

Set the path, where to store the config. Make sure you have full read/write access to the given directory. (Default: $XDG\_DATA\_HOME/signal-cli ($HOME/.local/share/signal-cli))

he password and cryptographic keys are created when registering and stored in the current users home directory, the directory can be changed with **--config**:

$XDG\_DATA\_HOME/signal-cli/ ($HOME/.local/share/signal-cli/)

For legacy users, the old config directories are used as a fallback:

$HOME/.config/signal/

$HOME/.config/textsecure/

## Sending a test message:

signal-cli -u <full\_international\_mobile\_number> send -m "This is a test message" <RECEIVER\_full\_international\_mobile\_number>

# Install required packages:

### Update the current packages / raspbian:

$ **sudo apt update**

$ **sudo apt full-upgrade -y**

$ **sudo reboot now**

$ **sudo apt clean**

$**mkdir -p ~/\_downloaded**

### Install latest JAVA JDK (v11.0.11+9-post-Raspbian-1deb10u1)

$ **sudo apt install default-jdk**  
$ **java --version**

pi@raspberrypi:~ $ java --version

openjdk 11.0.11 2021-04-20

OpenJDK Runtime Environment (build 11.0.11+9-post-Raspbian-1deb10u1)

OpenJDK Server VM (build 11.0.11+9-post-Raspbian-1deb10u1, mixed mode)

Add JAVA\_HOME to the environment file:

$ **sudo nano /etc/environment**

Add line: **JAVA\_HOME="/usr/lib/jvm/java-11-openjdk-armhf"**

**Ctrl-O** to write the file

**Ctrl-X** to close

### Install version v7.1 of Gradle

(<https://gradle.org/install/>)  
Copy down the binary-only from: <https://gradle.org/next-steps/?version=7.1&format=bin> :  
$ **wget -P ~/\_downloaded**[**https://services.gradle.org/distributions/gradle-7.1-bin.zip**](https://services.gradle.org/distributions/gradle-7.1-bin.zip)

$ **sudo mkdir /opt/gradle**  
$ **sudo unzip -d /opt/gradle ~/\_downloaded/gradle-7.1-bin.zip**

Add gradle to the PATH:  
$ **export PATH=$PATH:/opt/gradle/gradle-7.1/bin**

$ **sudo nano ~/.profile**

To the end of the file, add line: **PATH="$PATH:/opt/gradle/gradle-7.1/bin"** + <Enter>

**Ctrl-O** to write the file

**Ctrl-X** to exit

##### Check if all is ok by getting the version of Gradle:

$ **gradle -v**

Welcome to Gradle 7.1!

Here are the highlights of this release:

- Faster incremental Java compilation

- Easier source set configuration in the Kotlin DSL

For more details see https://docs.gradle.org/7.1/release-notes.html

------------------------------------------------------------

Gradle 7.1

------------------------------------------------------------

Build time: 2021-06-14 14:47:26 UTC

Revision: 989ccc9952b140ee6ab88870e8a12f1b2998369e

Kotlin: 1.4.31

Groovy: 3.0.7

Ant: Apache Ant(TM) version 1.10.9 compiled on September 27 2020

JVM: 11.0.11 (Raspbian 11.0.11+9-post-Raspbian-1deb10u1)

OS: Linux 5.10.17-v7+ arm

### Install Rust:

<https://www.rust-lang.org/tools/install>  
Run the below and follow instructions:  
$ **curl --proto '=https' --tlsv1.2 -sSf**[**https://sh.rustup.rs**](https://sh.rustup.rs/)**| sh**  
**Choose 1** Proceed with installation (default)

Once finished, add rust to the current Path  
$ **export PATH="$PATH:~/.cargo/bin"**

$ **sudo nano ~/.profile**

To the end of the file, add line: **PATH="$PATH:~/.cargo/bin"** + <Enter>

**Ctrl-O** to write the file

**Ctrl-X** to exit

##### Check rust is working by getting its version:

$ **rustup --version**

rustup 1.24.3 (ce5817a94 2021-05-31)

info: This is the version for the rustup toolchain manager, not the rustc compiler.

info: The currently active `rustc` version is `rustc 1.53.0 (53cb7b09b 2021-06-17)`

### Install protobuf-compiler:

$ **sudo apt install protobuf-compiler**

$ **sudo apt list -a protobuf-compiler**

Listing... Done

protobuf-compiler/stable,now 3.6.1.3-2+rpi1 armhf [installed]

### Install Git

Use one of the below methods. v2.20.1 worked ok, to build signal-cli v0.8.3 (as we are only using the basic methods like ‘pull’, ‘clone’ and ‘create branch’.)

Reference: <https://linuxize.com/post/how-to-install-git-on-raspberry-pi/#installing-git-from-the-source>

#### Install Git with apt-get (v2.20.1)

$ **sudo apt install git**

$ **git --version**

git version 2.20.1

#### Install latest stable version of Git from source (v2.32.0)

Start by installing the dependencies necessary to build Git on Raspbian:

$ **sudo apt install make libssl-dev libghc-zlib-dev libcurl4-gnutls-dev libexpat1-dev gettext**

Once the installation is complete, open your browser, visit the [Git project’s mirror](https://github.com/git/git/releases) on GitHub and copy the latest release link URL that ends in .tar.gz:

$ **cd ~/\_downloaded/**

$ **sudo wget** [**https://github.com/git/git/archive/v2.32.2.tar.gz -O git.tar.gz**](https://github.com/git/git/archive/v2.32.2.tar.gz%20-O%20git.tar.gz)

Next, extract the tarball and change to the git source directory by typing:

$ **sudo tar -xf git.tar.gz**

$ **cd git-\***

Run the following two commands to compile and install Git:

$ **sudo make prefix=/usr/local allsudo make prefix=/usr/local install**

Type git --version to verify the installation:

$ **git –version**

### Install zip and unzip

$ **sudo apt install zip unzip**

$ **sudo apt list -a zip**

Listing... Done

zip/stable,now 3.0-11 armhf [installed]

$ **sudo apt list -a unzip**

Listing... Done

unzip/stable,now 6.0-23+deb10u2 armhf [installed]

### Reboot

After installing all the packages, reboot the raspberry:

$ **sudo reboot now**

After the reboot, log back in and check the below 2 variables:

$ **echo $JAVA\_HOME**

Should be: /usr/lib/jvm/java-11-openjdk-armhf

$ **echo $PATH**

Should be: /home/pi/.cargo/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/usr/games:/opt/gradle/gradle-7.1/bin:~/.cargo/bin