

How to Uninstall Any Android App Using ADB (Including System Apps and Bloatware)

ADB is a powerful set of tools that give you more control over your Android device. Although ADB is intended for Android developers, you don't need any programming knowledge to uninstall Android apps with it.

Android devices come with a lot of pre-installed apps and while some of them can be useful, many of them are redundant and unwelcome. Luckily, you can uninstall any Android app unconditionally using ADB.

ADB is a powerful set of tools that give you more control over your Android device. Although ADB is intended for Android developers, you don't need any programming knowledge to uninstall Android apps with it.

ADB, or Android Device Bridge, is a tool that lets you run code on Android from your computer. ADB opens up a lot of possibilities through the power of the command line. However, you can also use ADB for simple tasks like using ADB to transfer your clipboard, remotely install apps, and of course uninstall them.

Uninstalling apps using ADB doesn't require root access, and it's pretty straightforward once you get the hang of it. In addition to letting you uninstall almost any app—including system apps and bloatware—you can also use ADB to uninstall apps remotely via Wireless Debugging.

Since ADB communicates with your device through code, you'll need to refer to the app's full package name, rather than the app's more familiar brand name, to uninstall it (this would be something like "com.facebook.katana" instead of Facebook, for example).

The package name may sound like something developers mess with, but the truth is you can easily find an app's package name with ADB. Once you have the package name, it only takes a single command to uninstall the app forever.

1. Install ADB on your computer

You need to install ADB on your computer before uninstalling the app on your Android device. You can install it by downloading it and then launching your computer's terminal in the ADB folder. Another way to install ADB on your computer is to use Terminal.

1. Download Platform Tools SDK (ADB)

<https://developer.android.com/tools/releases/platform-tools?hl=vi>

ADB is available on Windows, Mac, and Linux. If you're a Linux user, you probably don't need any instructions on how to install apps using the Terminal. On other platforms, you can use Scoop for Windows and Homebrew for Mac to install ADB. These two are command line installers that let you use the Terminal to install apps.

We will use a Windows computer as an example. Once Scoop is installed, you can install ADB with a single command via Command Prompt.

Launch Command Prompt by searching for cmd in the Start menu, then type the command below and press **Enter** on your keyboard.

```
scoop install adb
```

Scoop will automatically download and install ADB. If there are any missing dependencies, Scoop will ask you for permission to install them. Type **Y** and press **Enter** to allow the installation.

```
* eb1c276f6 aks-engine: Update to version 0.75.2 12 hours ago
* c3a940560 acorn: Update to version 0.5.0 12 hours ago
* b53db40d1 dua@2.19.0: Add 32bit variant (#4365) 15 hours ago
* ae1395415 teleport: Update to version 11.2.3 16 hours ago
* 4c3623c38 regula: Update to version 3.0.0 16 hours ago
* eec2bea3c pssh: Update to version 7.3.2 16 hours ago
* a1a6c365c pmpr: Update to version 7.26.0 16 hours ago
* 15cb8eeef mill: Update to version 0.10.11 16 hours ago
* 2ccb3e317 walker: Update to version 2.0.3 16 hours ago
* 2ba983035 jenkins: Update to version 2.388 16 hours ago
* bc13ec6a7 jdtls: Update to version 1.20.0-202301241650 16 hours ago
* 3bcbcf124 go-containerregistry: Update to version 0.13.0 16 hours ago
* e94dcb29 fzf: Update to version 0.37.0 16 hours ago
* 1fb841d31 fastfetch: Update to version 1.9.1 16 hours ago
* 3a3881bee aws-sam-cli: Update to version 1.71.0 16 hours ago
* cf55fcf7f aks-engine: Update to version 0.75.0 16 hours ago
Scoop was updated successfully!
Installing 'adb' (33.0.3) [64bit] from main bucket
Loading platform-tools_r33.0.3-windows.zip from cache
Checking hash of platform-tools_r33.0.3-windows.zip ... ok.
Extracting platform-tools_r33.0.3-windows.zip ... done.
Linking ~\scoop\apps\adb\current => ~\scoop\apps\adb\33.0.3
Creating shim for 'adb'.
Creating shim for 'dtracedump'.
Creating shim for 'etctool'.
Creating shim for 'fastboot'.
Creating shim for 'hprof-conv'.
'adb' (33.0.3) was installed successfully!
C:\Users\Amir>
```

Now ADB is installed and ready to serve you. The ADB package name is different on Homebrew and Linux. You can install ADB using Homebrew on your Mac with:

```
brew install android-platform-tools
```

And on Linux you can use:

```
sudo apt-get install android-tools-adb
```

2. Connect ADB to Android device

Of course, you need to enable USB Debugging on your Android to use ADB with that device. This allows you to connect to your device via USB cable or wirelessly.

If your device is a smartphone, then the easiest way is to plug it into your computer using a USB cable. If you are dealing with a stationary device like an Android TV box, then you should try Wireless Debugging with ADB.

Regardless of which method you choose to connect, you can get a list of connected devices by running the command below in Terminal:

```
adb devices
```

```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Amir>adb devices
List of devices attached
4cce8048 device

C:\Users\Amir>
```

This command displays the serial number for devices connected via USB and the local IP address for wireless connections.

3. Find the application package name with ADB

Although most apps have simple labels, Android doesn't use the app's label to identify them. What if two apps have the same label? To avoid potential confusion, Android uses a unique package name to identify the app. For example, Google Maps appears as **Maps** in the app list, but the actual name of Google Maps is **com.google.android.apps.maps**.

Luckily, you can use ADB in the same Terminal window to find the package name of an app. Once ADB is connected to your device, run the command below to enter the ADB shell:

```
adb shell
```

You should now see your device name next to the flashing indicator. Next, run the command below to get a list of all the packages installed on your Android device:

```
pm list packages
```

```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [Version 10.0.22621.1185]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Amir>adb shell
OnePlus6T:/ $ pm list packages
package:me.turbovpn.vpn
package:com.android.internal.display.cutout.emulation.noCutout
package:com.google.android.networkstack.tethering
package:com.oneplus.calculator
package:net.oneplus.weather
package:com.asana.app
package:com.skype.raider
package:com.android.cts.priv.ctsshim
package:com.samsung.android.gearoplugin
package:com.qualcomm.qti.qms.service.telemetry
package:com.google.android.youtube
package:com.oneplus.gamespace
package:com.qualcomm.qti.qcolor
package:com.android.internal.display.cutout.emulation.corner
package:com.google.android.ext.services
package:com.qualcomm.qti.improvetoast.service
package:com.android.internal.display.cutout.emulation.double
package:com.microsoft.appmanager
package:com.bama.consumer
package:com.android.providers.telephony
package:com.oneplus.commonoverlay.com.google.android.networkstack
package:com.android.dynsystem
package:com.google.android.googlequicksearchbox
package:com.google.android.cellbroadcastservice
package:com.android.providers.calendar
```

Call the ADB package manager to list all the packages installed on your Android device. The sheer length of this list – and the packages you've never heard of – should suggest that there are a lot of apps installed on your device that you don't even know about. Most of these are background system apps that silently keep your Android device running.

Warning :

You can uninstall any of the listed Android packages using ADB, but that doesn't mean you should. Most of these apps are essential to your system running, and uninstalling them could potentially affect the functionality of your device. Only uninstall system apps if you know what you're doing!

A complete list of installed packages on your Android device can help you understand better, but searching for the package name of a specific app won't be very effective.

You can use the command below to filter the results:

```
pm list packages |grep chrome
```



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Amir>adb shell
OnePlus6T:/ $ pm list packages |grep chrome
package:com.android.chrome
OnePlus6T:/ $
```

This command filters the package list to only include packages named *chrome*. Replace *chrome* with your application name or developer name to find packages for that application.

4. Uninstall the app

Now that you have the package name of the app, the app is completely in your hands. If you are still in the ADB shell, exit that shell by typing `exit` and pressing **Enter**. Then enter the package name in the command below and run it:

```
adb uninstall com.spotify.lite
```



```
C:\WINDOWS\system32\cmd. x + v
C:\Users\Amir>adb uninstall com.spotify.lite
Success
C:\Users\Amir>
```

This command will uninstall Spotify Lite from the connected Android device. Replace **com.spotify.lite** with the package name of the app you want and it will disappear after ADB outputs a **Success result**.

Note that the above command will only ask the system to uninstall the app for the current user. There is no way to uninstall the app from all users unless you root your phone.

Be warned, uninstalling system apps can potentially brick your device, so only uninstall apps you're sure about. Apps like Gmail, Google Play Music, Google Play Movies, etc. can be safely uninstalled, but never delete the Google Play Store or any files associated with it. If your phone becomes unstable after uninstalling a particular app, reinstall it from the Google Play Store or factory reset your phone.

Depending on the manufacturer, some phones only come with a few bloatware and system apps that can be disabled easily, while others are loaded with third-party apps that users cannot delete or disable.

Android users have more freedom due to its open source nature, however, Android still has some security restrictions. You cannot uninstall system apps and most of the pre-installed apps on your Android device without taking extra steps. ADB is a tool that can circumvent this restriction. It is your phone after all, and you should have full control over it. The instructions above will allow you to remove those bloatware apps from your Android phone without having to root your phone.

The best thing about uninstalling Android apps using ADB is that you don't need to root your device to use it. All you have to do is connect your device, find the package name of the app, and run the uninstall command. Now you know how to do all that. Say goodbye to bloatware and apps you don't use!

You finished reading the article "**How to Uninstall Any Android App Using ADB (Including System Apps and Bloatware)**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.