

[Questions](#) [Tags](#) [Users](#) [Badges](#) [Unanswered](#) [Ask](#)

## Trying to start a service on boot on Android

181

android

service

broadcastreceiver

I've been trying to start a service when a device boots up on android, but I cannot get it to work. I've looked at a number of links online but none of the code works. Am I forgetting something?

### AndroidManifest.xml

```
<receiver
  android:name=".StartServiceAtBootReceiver"
  android:enabled="true"
  android:exported="false"
  android:label="StartServiceAtBootReceiver" >
  <intent-filter>
    <action android:name="android.intent.action._BOOT_COMPLETED" />
  </intent-filter>
</receiver>

<service
  android:name="com.test.RunService"
  android:enabled="true" />
```

### BroadcastReceiver

```
public void onReceive(Context context, Intent intent) {
  if ("android.intent.action.BOOT_COMPLETED".equals(intent.getAction())) {
    Intent serviceLauncher = new Intent(context, RunService.class);
    context.startService(serviceLauncher);
    Log.v("TEST", "Service loaded at start");
  }
}
```

Thanks

[share](#) [improve this question](#)**Alex**

964 ● 3 ● 10 ● 9

**Asked**

May 6 '10 at 20:56

**Greg**

9,134 ● 10 ● 36 ● 64

**Edited**

Feb 24 at 12:12

1

i dunno what i did but i think it works now it might have been the android.permission="android.permission.RECEIVE\_BOOT\_COMPLETED" for the receiver – [Alex May 6 '10 at 22:58](#)

have you checked the extra "\_" in <action android:name="android.intent.action.\_BOOT\_COMPLETED"/>

22 You should accept the answer if you found a valuable suggestion which helped you fix your problem... This would help you in turn to improve your accept ratio, and even help others who may face a similar issue in the future! – [Mahendra](#) Dec 6 '11 at 11:42

6 Please accept Timo's answer. – [Nemanja Boric](#) Nov 7 '12 at 20:33

You ask questions and accept none. Is this behavior by habit or choice? – [Siddharth](#) Sep 24 at 5:08

[add a comment](#)

10 Answers

order by [votes](#)

390 The other answers look good, but I thought I'd wrap everything up into one complete answer.

You need the following in your AndroidManifest.xml file:

1) In your <manifest> element:

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
```

2) In your <application> element (be sure to use a fully-qualified [or relative] class name for your BroadcastReceiver):

```
<receiver android:name="com.example.MyBroadcastReceiver">
  <intent-filter>
    <action android:name="android.intent.action.BOOT_COMPLETED" />
  </intent-filter>
</receiver>
```

(you don't need the android:enabled, exported, etc., attributes... the Android defaults are correct)

In MyBroadcastReceiver.java:

```
package com.example;

public class MyBroadcastReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        Intent startServiceIntent = new Intent(context, MyService.class);
        context.startService(startServiceIntent);
    }
}
```

From the original question:

- it's not clear if the <receiver> element was in the <application> element

- it's not clear if the correct fully-qualified (or relative) class name for the BroadcastReceiver was specified
- there was a typo in the <intent-filter>

[share](#) [improve this answer](#)



**Timo Bruck**

4,853 ●2●8●11

**Answered**

Mar 25 '11 at 23:55



**Alexis**

3,059 ●4●41●79

**Edited**

Nov 4 '13 at 5:15

2 This looks good. I'm gonna use this as a basis, thanks :). No checkmark or upvotes or response sadly : (. Anyone verify this? – [Nanne Apr 18 '11 at 18:18](#)

26 Just a complement: make sure your app is install in internal memory <manifest xmlns:android="..." package="..." android:installLocation="internalOnly"> – [Bao Le Aug 9 '11 at 2:00](#)

Its very simple and perfect, Just i saw the answer once , i wont forgot ever. – [Kartihkraj Duraisamy Jan 25 '13 at 6:38](#)

2 In Android Jellybean 4.2.2 in the <receiver> tag I had to use the class' relative name instead of the fully-qualified name for the service to start, as noted in [stackoverflow.com/questions/16671619/...](http://stackoverflow.com/questions/16671619/) – [Piovezan May 21 '13 at 18:38](#)

5 If the receiver is used for different stuff:<br> if ("android.intent.action.BOOT\_COMPLETED".equals(intent.getAction())) { Intent serviceIntent = new Intent(context, Service\_Location.class); // i.putExtra("KEY1", "Value to be used by the service"); context.startService(serviceIntent); } – [Gunnar Bernstein Nov 30 '13 at 10:51](#)

[show 2 more comments](#)

58 As an additional info: BOOT\_COMPLETE is sent to applications *before* external storage is mounted. So if application is installed to external storage it won't receive BOOT\_COMPLETE broadcast message.

More details [here](#) in section **Broadcast Receivers listening for "boot completed"**

[share](#) [improve this answer](#)



**inazaruk**

40.3k ●10●116●125

**Answered**

Nov 22 '10 at 13:59

23 Along with

```
<action android:name="android.intent.action.BOOT_COMPLETED" />
```

also use,

```
<action android:name="android.intent.action.QUICKBOOT_POWERON" />
```

HTC devices dont seem to catch BOOT\_COMPLETED

[share](#) [improve this answer](#)



**Tony**  
350 ● 3 ● 9

**Answered**  
May 14 '13 at 11:52

17

note that at the beginning of the question, there is a typo mistake:

```
<action android:name="android.intent.action._BOOT_COMPLETED"/>
```

instead of :

```
<action android:name="android.intent.action.BOOT_COMPLETED"/>
```

one small "\_" and all this trouble :)

[share](#) [improve this answer](#)



**Evgeny Erlihman**  
258 ● 3 ● 8

**Answered**  
Feb 5 '11 at 9:19

**Edited**  
Feb 5 '11 at 12:10

10

I found out just now that it might be because of Fast Boot option in Settings > Power

When I have this option off, my application receives a this broadcast but not otherwise.

By the way, I have Android 2.3.3 on HTC Incredible S.

Hope it helps.

[share](#) [improve this answer](#)



**Omer Akhter**  
143 ● 1 ● 7

**Answered**  
Jan 11 '12 at 20:21

Definitely possible cause of the issue. Observed also on HTC Desire C running Android 4.0.3. – [Zelimir Jan 8 '13 at 16:32](#)

that solved my problem – [Abhishek Chauhan Mar 2 '13 at 18:34](#)

that's y u should look this [stackoverflow.com/a/16542461/1239006](http://stackoverflow.com/a/16542461/1239006) – [MatheusJardimB Apr 26 at 2:32](#)

[add a comment](#)

How to start service on device boot(autorun app, etc.)

For first: since version Android 3.1+ you don't receive BOOT\_COMPLETE if user never started your app at least once or user "force closed" application. This was done to prevent malware automatically register service. This security hole was closed in newer versions of Android.

Solution:

Create app with activity. When user run it once app can receive BOOT\_COMPLETE broadcast message.

For second: BOOT\_COMPLETE is sent before external storage is mounted. if app is installed to external storage it won't receive BOOT\_COMPLETE broadcast message.

In this case there is two solution:

1. Install your app to internal storage
2. Install another small app in internal storage. This app receives BOOT\_COMPLETE and run second app on external storage.

If your app already installed in internal storage then code below can help you understand how to start service on device boot.

In Manifest.xml

Permission:

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
```

Register your BOOT\_COMPLETED receiver:

```
<receiver android:name="org.yourapp.OnBoot">
  <intent-filter>
    <action android:name="android.intent.action.BOOT_COMPLETED"/>
  </intent-filter>
</receiver>
```

Register your service:

```
<service android:name="org.yourapp.YourCoolService" />
```

In receiver OnBoot.java:

```

public class OnBoot extends BroadcastReceiver
{

    @Override
    public void onReceive(Context context, Intent intent)
    {
        // Create Intent
        Intent serviceIntent = new Intent(context, YourCoolService.class);
        // Start service
        context.startService(serviceIntent);

    }

}

```

---

For HTC you maybe need also add in Manifest this code if device don't catch RECEIVE\_BOOT\_COMPLETED:

```
<action android:name="android.intent.action.QUICKBOOT_POWERON" />
```

Reciever now look like this:

```

<receiver android:name="org.yourapp.OnBoot">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED"/>
        <action android:name="android.intent.action.QUICKBOOT_POWERON" />
    </intent-filter>
</receiver>

```

---

How to test BOOT\_COMPLETED without restart emulator or real device? It's easy. Try this:

```
adb -s device-or-emulator-id shell am broadcast -a android.intent.action.BOOT_COMPLETED
```

How to get device id? Get list of connected devices with id's:

```
adb devices
```

adb in ADT by default you can find in:

```
adt-installation-dir/sdk/platform-tools
```

Enjoy! )

---

[share](#) [improve this answer](#)



**user3439968**  
353 ● 1 ● 5

**Answered**  
Sep 24 at 21:04

---

**Edited**  
Sep 24 at 21:34

8

I think your manifest needs to add:

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
```

[share](#) [improve this answer](#)



**RickNotFred**

2,362 ●1 ●12 ●25

**Answered**

May 6 '10 at 21:14

1

i forgot to mention i have that – [Alex May 6 '10 at 21:18](#)

Move it to receiver tag - look at example i've posted below. – [Alex Volovoy May 6 '10 at 21:55](#)

@AlexVolovoy what happened to the example? – [naxa Oct 30 '13 at 12:05](#)

[add a comment](#)

3

Refer This Link <http://khurramitdeveloper.blogspot.in/2013/06/start-activity-or-service-on-boot.html> Step by Step procedure to use boot on Service

[share](#) [improve this answer](#)



**Jaichander**

111 ●2 ●12

**Answered**

Feb 17 at 4:24

2

I have an additional <category>-tag, don't know if that makes any difference.

```
<receiver android:name="BootIntentReceiver">
  <intent-filter>
    <action android:name="android.intent.action.BOOT_COMPLETED" />
    <category android:name="android.intent.category.HOME" />
  </intent-filter>
</receiver>
```

Have you tried ommiting the if-clause

"android.intent.action.BOOT\_COMPLETED".equals(intent.getAction()), as the receiver probably only receives that intent anyway?

[share](#) [improve this answer](#)



**Nick**

1,158 ●13 ●54

**Answered**

May 6 '10 at 21:12

tried this and it didn't work btw i forgot to mention i also have the <uses-permission android:name="android.permission.RECEIVE\_BOOT\_COMPLETED"/> – [Alex May 6 '10 at 21:18](#)

2

just in case: adding `android.intent.category.HOME` to any tag in the `AndroidManifest` will cause the Samsung Galaxy Tab to run the app in compatibility mode, even after using the hack to turn compatibility mode off. not sure if this is the same for other tabs. i recommend not setting the HOME category at all. it's unnecessary. – [moonlightcheese](#) Mar 10 '11 at 23:31

[add a comment](#)

1

In fact,I get into this trouble not long ago,and it's really really easy to fix,you actually do nothing wrong if you setup the "`android.intent.action.BOOT_COMPLETED`" permission and intent-filter.

Be attention that if you On Android 4.X,you have to run the broadcast listener before you start service on boot,that means,you have to add an activity first,once your broadcast receiver running,your app should function as you expected,however,on Android 4.X,I haven't found a way to start the service on boot without any activity,I think google did that for security reasons.

[share](#) [improve this answer](#)



**David Hx**  
76 ● 1 ● 9

**Answered**  
Aug 20 '13 at 14:35

[meta](#) [chat](#) [tour](#) [help](#) [blog](#) [privacy policy](#) [legal](#) [contact us](#) [full site](#)

Download the Stack Exchange Android app

2014 stack exchange, inc