

I thought there is a quick fix for this problem - just set targetSdkVersion to 27, but Google Play Console doesn't allow to upload such APK anymore: "Upload failed. Your app currently targets there is currently no solution at all, please increase the bug priority!

sh...@gmail.com <sh...@gmail.com> #6

I build a Pixel XL ROM from source android-10.0.0_r20 and debug. Then I found something interesting.

private boolean isCameraPermissionGranted(String packageName, int uid) {

When App target to API 28 and above. The function 'private boolean isCameraPermissionGranted(String packageName, int uid)' in file 'services/usb/java/com/android/server/usb/UsbUserS& That is because line 150 'mUserContext.checkCallingPermission(android.Manifest.permission.CAMERA)' always return PackageManager.PERMISSION_DENIED;

```
int targetSdkVersion = android.os.Build.VERSION_CODES.P;
      ApplicationInfo alnfo = mPackageManager.getApplicationInfo(packageName, 0);
      // compare uid with packageName to foil apps pretending to be someone else
      if (alnfo.uid != uid) {
         Slog.i(TAG, "Package" + packageName + " does not match caller's uid " + uid);
        return false;
      targetSdkVersion = aInfo.targetSdkVersion;
    } catch (PackageManager.NameNotFoundException e) {
      Slog.i(TAG, "Package not found, likely due to invalid package name!");
      return false;
    if (targetSdkVersion >= android.os.Build.VERSION_CODES.P) {
      int allowed = mUserContext.checkCallingPermission(android.Manifest.permission.CAMERA);
      if (android.content.pm.PackageManager.PERMISSION_DENIED == allowed) {
        Slog.i(TAG, "Camera permission required for USB video class devices");
        return false;
    }
    return true;
The result of 'pid != Process.myPid()' (In file 'core/java/android/app/ContextImpl.java' line 1863) always false in function 'checkCallingPermission'.
This line is in function:
  @Override
  public int checkCallingPermission(String permission) {
    if (permission == null) {
```

So that means Binder.getCallingPid() is equal with Process.myPid() and the function always return PackageManager.PERMISSION_DENIED;

And why Binder.getCallingPid() is equal with Process.myPid()? I think I have found the answer.

throw new IllegalArgumentException("permission is null");

 $return\ check Permission (permission, pid, Binder.get Calling Uid ());$

int pid = Binder.getCallingPid();
if (pid != Process.myPid()) {

return PackageManager.PERMISSION_DENIED;

This function is in file 'services/usb/java/com/android/server/usb/UsbService.java' line 350 @Override
public boolean hasDevicePermission(UsbDevice device, String packageName) {
 final int uid = Binder.getCallingUid();
 final int userId = UserHandle.getUserId(uid);

final long token = Binder.clearCallingIdentity();

return getSettingsForUser(userId).hasPermission(device, packageName, uid);
} finally {
Binder.restoreCallingIdentity(token);
}
}
It calls 'Binder.clearCallingIdentity();' which will make Binder.getCallingPid() equal with Process.myPid().

It calls Binder.clearCallingIdentity(); which will make Binder.getCallingPid() equal with Process.myPid()

SO whether or not Camera permission is added, USB permission will never be granted.

The attachments contain more debug infos.



0 B ②

deleted 0 B ②

sh...@gmail.com <sh...@gmail.com>_#7

This issue can be fixed by adding the patch https://android-review.googlesource.com/c/platform/frameworks/base/+/1193928
Test it on Pixel XL with AOSP tags android-10.0.0_r25 and android-10.0.0_r20.

he@gmail.com <he@gmail.com><u>#8</u></he@gmail.com>
We have a custom usb camera, with a custom usb app and hundreds of users. Our users on Android 10 cannot use the app anymore. Since Play store won't allow us to target SDK 27 there is I can't understand why this is not getting more attention?
re@gmail.com <re@gmail.com><u>#9</u></re@gmail.com>
This issue is also important for my team. Please work on it and solve asap.
fr@500k.com.br <fr@500k.com.br> <u>#10</u></fr@500k.com.br>
This is also important for us. Approximately 3.000 users have been affected by this issue. We are about to lose 2 contracts if we don't find an alternative.
st@gmail.com <st@gmail.com><u>#11</u></st@gmail.com>
Also affecting our application - hopefully Google will choose lower the Google Play requirements to 27 - I see no way to make sure all the android 10 devices will be updated with the fix.
st@gmail.com <st@gmail.com><u>#12</u></st@gmail.com>
Hi Guys, small update about the issue - I see this patch was applied in android-10.0.0_r30 here: abe984f550e6 (Nov 13 15:47:42 2019 -0800) - and already available on my Pixel 4 with latest s
sh@gmail.com <sh@gmail.com> #13</sh@gmail.com>
Yes, now this issue can be fixed by the March 2020 Android security patch (2020-03-05 patch level).
ge@gmail.com <ge@gmail.com> <u>#14</u></ge@gmail.com>
Is there any workaround if the phone my app is running on doesn't have the latest security update?
ma@gmail.com <ma@gmail.com><u>#15</u></ma@gmail.com>
hi, we had already installed the Android security patch May 5th, 2020. but we also have this issue on my ASUS ZenPhone5Z ASUS_Z01RD(android-10) . can anyone tell us how to fix it, thanks.
O B ③
ma@gmail.com <ma@gmail.com><u>#16</u></ma@gmail.com>
We have seen that the security patch released in March(2020-03-05) solved the problem for some devices(such as Pixel series). May I ask when will you release a security patch that can solve
[Deleted User] <[Deleted User]>#17
Hi, Is this issue solved, In OnePlus7T with latest security patch(2020-05-01), it is not working Do we have any solution for this, guys please help
ma@gmail.com <ma@gmail.com><u>#18</u></ma@gmail.com>
Hi, Is this issue solved, In many devices (Sony, Sharp, Samsung, Oppo, Huawei, LG, ASUS), these are not working. May I ask when will you release a security patch that can solve this problem on these devices?
he@slickorange.co.za <he@slickorange.co.za> #19</he@slickorange.co.za>
I have a Xiaomi Mi A2 which is on the Android One program - My app is still not working, even on this device with "stock" Android. I know moaning is not going to help but this is not great, what makes it even worse is that this issue was reported during Beta but simply ignored.
bi@gmail.com <bi@gmail.com> #20</bi@gmail.com>
nassime
op@gmail.com <op@gmail.com> #21</op@gmail.com>
The problem is not solved. USB permission dialog still not appears on some devices even with recent security patch. Example: Motorola Moto g7 plus XT1965-3, Android 10, Security update:
da@gmail.com <da@gmail.com> #22</da@gmail.com>

We're reliant on this for running medical imaging devices, it's vitally important for us that this gets fixed.
ni@gmail.com <ni@gmail.com> #23</ni@gmail.com>
Seen this on every device I have running Android 10. Why has such useful and basic support stayed broken for so long?
st@gmail.com <st@gmail.com> #24</st@gmail.com>
Guys,
I don't think Google can do anything with devices of other manufactures. They merged the patch to android 10 branch and Pixel 4 & Samsung Galaxy S10 (but not S10+) already got the updat
So it's better to address those questions to the devices manufacturers than post here - that will be much effective.
Thank you
am@google.com <am@google.com> #25</am@google.com>
Hi,
Could you please check with the latest Android release and share the status and reports?
je@gmail.com <je@gmail.com> <u>#26</u></je@gmail.com>
Just to elaborate on this: there are two different things at play here: AOSP version and security patches.
As stated previously, cothe fix is available starting with the tag and roid-10.0.0_30, but is not included in any security patches* (because it's not a security concern). So if a manufacturer in 10.0.0_30.
The reason why it works for Google Pixel series is because Google publishes updates that include the latest versions of the AOSP (>= android=10. 0. 0_30), not just security patches.
Conclusion: just ignore the security patch level, this is not the important information. What really matters is the tag of the AOSP used.
*: Earlier this year I checked that none of the security patches from November 2019 to April 2020 included the fix. Newer patches could include it.
kr@bandyer.com <kr@bandyer.com>_#27</kr@bandyer.com>
Hi,
That sure clears out why it's still broken on other manufacturers.
For my company use case is important to have this working. Is there a fix that we can employ in such cases, even something in JNI? Because we all know the manufacturers will not update soon their devices if they ever will do it.
st@gmail.com <st@gmail.com> #28</st@gmail.com>
Hi,
I tried to prepare a workaround, but found it's impossible with following the Android security model - the OS just not notifying the application (you can check the patch for more information) a
Workarounds: * Downgrade the application AndroidManifest.xml android:targetSdkVersion to 27 - will not allow to upload to google play due to their security restrictions, but at least you could send apk to
* Probably the workaround could be prepared for rooted devices, just put newly connected usb devices to rw mode. Not great for customers. Hopefully this will help.
lu@gmail.com <lu@gmail.com> #29</lu@gmail.com>
Does not work yet for me on galaxy note 10 plus. How do I know what version of AOSP am I running?
ma@gmail.com <ma@gmail.com><u>#30</u></ma@gmail.com>
These are not working. Sharp Android One S3
Sharp Android One S5
Sharp Android One S7 Sharp Android One X4
Sharp AQUOS R Compact
Sharp AQUOS R2 Sharp AQUOS R2 compact
Sharp AQUOS R3
Sharp AQUOS R5G Sharp AQUOS sense3
Sharp AQUOS sense3 plus sound
Sharp AQUOS zero Sharp AQUOS zero2

guys please help.

je@gmail.com <je@gmail.com> #31</je@gmail.com>
How do I know what version of AOSP am I running?
For Pixel devices:
1. Go to Android settings -> About phone -> Build number 2. The build number looks something like QP1A. 192005. 007. A3 (this is the build number of my Pixel XL (1st gen) as of 2020/04/20) 3. Refer to the section "Source code tags and builds" in the page ←> Codenames, Tags, and Build Numbers 4. Look for your build number in the list 5. If the corresponding tag to your build number is >= android-10. 0. 0_r30, the phone has the fix. Otherwise it does not 6. For instance, my Pixel XL's build number QP1A. 192005. 007. A3 maps to android-10. 0. 0_r17: the bug is still there For other manufacturers, I guess it depends of their branch management.
di@gmail.com <di@gmail.com>#32</di@gmail.com>
Спасибо, установил 27ой, камера заработал.
el@gmail.com <el@gmail.com>#33</el@gmail.com>
Downgrading all of my customers devices now. Waiting is no longer an option.
ch@google.com <ch@google.com><u>#34</u></ch@google.com>
Hello team,
Friendly ping. Please let us know if there is any update for the issue.
Thanks.
ez@gmail.com <ez@gmail.com>_#35</ez@gmail.com>
Google Play limit Target Version To 29 and it makes the bug more seriesly
zi@gmail.com <zi@gmail.com> #36</zi@gmail.com>
https://docs.google.com/document/d/1xen6NDCRRZ0r8JQCULc5espKqG2sp89JZQG-Nc9IEr4/edit?usp=drive_web (Gmail)
op@gmail.com <op@gmail.com> #37</op@gmail.com>
I suppose there will be no important updates for this issue anymore. 1. Google already fixed this bug in some revision of Android 10. 2. If the manufacturer performs complete Android updates, your device must get this fix and work properly. 3. If the manufacturer performs only security updates, your device can get this fix only after updating to Android 11. So it depends totally from manufacturers now, not from Google.
sh@gmail.com <sh@gmail.com> <u>#38</u></sh@gmail.com>
Well, Google has bring the issue back on android-10.0.0_r46, so it seems the issue unable to be fixed on Android 10.
tb@gmail.com <tb@gmail.com><u>#39</u></tb@gmail.com>
Can someone confirm whether this issue affects Camera2 API also? I want to purchase some high end USB cameras, but if they are not usable on Android, even with Camera2 API, then it wo
je@gmail.com <je@gmail.com> #40</je@gmail.com>
То сэ <u>#38</u> :
Well, Google has bring the issue back on android-10.0.0_r46, so it seems the issue unable to be fixed on Android 10.
Indeed it's possible to check that the latest tags for Android 10 published on co Codenames, Tags, and Build Numbers do not have the fix. I checked whether the fix is present for all the tags
1. All tags before android-10. 0. 0_r30: absent 2. All tags between android-10. 0. 0_r30 and android-10. 0. 0_r45: present 3. All tags between android-10. 0. 0_r46 and android-security-10. 0. 0_r50 (the last one at the moment): absent
The last tags do not have it because they are only backports of security patches (this can be checked in Android Security Backport Released). They probably are based on a tag older than
ya@gmail.com <ya@gmail.com> <u>#41</u></ya@gmail.com>

As someone mentioned before, the patch to fix this issue was reverted in the tag android-10.0.0_r46, from tag android-security-10.0.0_r48 to android-security-10.0.0_r54 are security release I just notice the patch was included again in android-11.0.0_r1 and it's also present in the current latest version android-11.0.0_r37.
Tags a builds: https://source.android.com/setup/start/build-numbers#source-code-tags-and-builds
Fix commit: https://cs.android.com/android/_/android/platform/frameworks/base/+/f7824ac70d3f6d731576b1d7d6172eb3a9ea0921
kj@gmail.com <kj@gmail.com> <u>#42</u></kj@gmail.com>
What is the point of a beta release if they don't fix showstopper bugs that are identified in it?
am@google.com <am@google.com><u>#43</u></am@google.com>
@shenya
Could you please let us know the current issue status with the new release "11.0.0 (RQ3A.210605.005, Jun 2021)" as we don't see the issue with this? Just for reference, here is the images link: https://developers.google.com/android/images
ad@google.com <ad@google.com><u>#44</u></ad@google.com>
Please provide the information requested in comment #43
kj@gmail.com <kj@gmail.com> #45</kj@gmail.com>
We need a fix for 10.x, not 11.x. Fix it in 10.x and lean hard on carriers to push the fix to all devices.
ad@google.com <ad@google.com><u>#46</u></ad@google.com>
We are still waiting for information requested in comment #43
kj@gmail.com <kj@gmail.com> <u>#47</u></kj@gmail.com>
Again, that request is irrelevant. We need a backport fix for deployed 10.x devices. By refusing to fix major issues in deployed versions, Google increases risks for commercial customers and
na@gmail.com <na@gmail.com> #48</na@gmail.com>
Any updates on this issue? It seems like it's completely ignored
ma@gmail.com <ma@gmail.com><u>#49</u></ma@gmail.com>
Thank You
jc@gmail.com <jc@gmail.com><u>#50</u></jc@gmail.com>
I don't understand much of what has been posted here. All I know is I have a Galaxy S9+ running android 10, up to date with the latest os provoded by Samsung. USB camera/software will no
kj@gmail.com <kj@gmail.com> <u>#51</u></kj@gmail.com>
 TL;DR: Google didn't fix the issue when it was identified in the Android 10 beta. They didn't fix it after it was widely deployed. Now they're talking about fixing it in Android 11 instead. Time to
ad@google.com <ad@google.com><u>#52</u></ad@google.com>
Please share the following info :
 Camera app info. Is it downloaded from Google play or developed by you (provide more details)? Build version on Pixel phone, the current build version with which you are facing the issue. Please provide bug report if available. Other phone OEMs (if any) with which you are facing the issue.
kj@gmail.com <kj@gmail.com> <u>#53</u></kj@gmail.com>
Irrelevant. The issue was already identified and fixed. Backport the fix and push it to ALL 10.x devices. At your expense if necessary.
de@dentulu.com <de@dentulu.com><u>#54</u></de@dentulu.com>
From November 2021, we must target to SDK level 30, so any update on this issue?

am@google.com <am@google.com> #55</am@google.com>
This should work on newer Android Version
Based on the info from SDK Platform release notes (https://developer.android.com/studio/releases/platforms) The API level 30 is Android 11.
We checked with the Sharp sense5G(Android 11) & Samsung S20(Android 11) then downloading the Camera APP(USB Camera) from google play, the USB camera works.
So the intended behavior is USB camera can work in Android 11(API level 30).
kj@gmail.com <kj@gmail.com> <u>#56</u></kj@gmail.com>
Unacceptable. You will fix all Android 10 devices in the wild.
am@gmail.com <am@gmail.com><u>#57</u></am@gmail.com>
Hab arms are unlasted
Usb camera unlocked
vi@google.com <vi@google.com><u>#58</u></vi@google.com>
Status: Won't Fix (Intended Behavior)
We have looked into the issue you have reported and would like to inform that this is working as intended. The USB camera can work in Android 11 (API level 30) and the fix is not available for
we have looked into the issue you have reported and would like to inform that this is working as intended. The obbigainera can work in Android 11 (Ar Fiever 50) and the fix is not available to
kj@gmail.com <kj@gmail.com> <u>#59</u></kj@gmail.com>
Bullshit. Fix it in Android 10 and lean on carriers to deploy the fix to all Android 10 devices in the wild at Google's expense.
building. The fill Android To did real on carriers to deploy the fix to all Android To devices in the wild at deolgie's expense.
ha@gmail.com <ha@gmail.com><u>#60</u></ha@gmail.com>
Please help me
is@gmail.com <is@gmail.com>#61</is@gmail.com>
D.
is@gmail.com <is@gmail.com> #62</is@gmail.com>
Fuuufudy
ad Compiler on and Compiler on #60
ad@gmail.com <ad@gmail.com> #63</ad@gmail.com>
مش شغالين عندي في برنامج بيعرض صوت من غير صورة usb camra و Esay cap برنامج
be@gmail.com <be@gmail.com> #64</be@gmail.com>
be@gman.com ve@gman.com> <u>#04</u>
Connect the camera to android10 phone
ha@gmail.com <ha@gmail.com>#65</ha@gmail.com>
Unacceptable. You will fix all Android 10 devices in the wild.
ha Camail ann tha Camail ann #66
ba@gmail.com <ba@gmail.com> #66</ba@gmail.com>
cdlm google!
ja@gmail.com <ja@gmail.com> <u>#67</u></ja@gmail.com>
Jaydeep