Can't start activity from BroadcastReceiver on android 10

Asked 3 years, 9 months ago   Modified 2 months ago   Viewed 17k times

▲

12

▼

🔖

🕘

I updated my OS version to android 10 last night, and since then the startActivity function inside the broadcast receiver is doing nothing. This is how I try to start the activity based on the answer of CommonsWare:

```
Intent i = new Intent(context, AlarmNotificationActivity.class);
i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q) { // This is at least android
10...

                Log.d("Debug", "This is android 10");
                // Start the alert via full-screen intent.
                PendingIntent startAlarmPendingIntent =
PendingIntent.getBroadcast(context, 0, i, PendingIntent.FLAG_UPDATE_CURRENT);
                String CHANNEL_ID = "my_channel_02";
                NotificationChannel channel = new NotificationChannel(CHANNEL_ID,

    context.getString(R.string.notification_channel_name_second),
                        NotificationManager.IMPORTANCE_HIGH);
                NotificationManager notificationManager = (NotificationManager)
context.getSystemService(Context.NOTIFICATION_SERVICE);
                notificationManager.createNotificationChannel(channel);
                NotificationCompat.Builder builder = new
NotificationCompat.Builder(context, CHANNEL_ID)
                    .setContentTitle("Um, hi!")
                    .setAutoCancel(true)
                    .setPriority(NotificationCompat.PRIORITY_HIGH)
                    .setFullScreenIntent(startAlarmPendingIntent, true);
                Log.d("Debug", "Try to load screen");
                notificationManager.notify(0, builder.build());

        }
```

The log shows that I am getting to the notify command but nothing happens. I am asking for USE_FULL_SCREEN_INTENT permission on the manifest so I should be able to use full-screen intents. My app is useless now because of that issue. Does anyone know how to solve it?

android   broadcastreceiver   android-pendingintent   start-activity   android-10.0

Share Improve this question Follow

3 Answers

Sorted by: Highest score (default) ⇕

▲

14

▼

🔖

✔

🕘

Android 10's restriction on background activity starts was announced about six months ago. You can read more about it in the documentation.

Use a high-priority notification, with an associated full-screen `Intent`, instead. See the documentation. This sample app demonstrates this, by using `WorkManager` to trigger a background event needing to alert the user. There, I use a high-priority notification instead of starting the activity directly:

```
val pi = PendingIntent.getActivity(
    appContext,
    0,
    Intent(appContext, MainActivity::class.java),
    PendingIntent.FLAG_UPDATE_CURRENT
)

val builder = NotificationCompat.Builder(appContext, CHANNEL_WHATEVER)
    .setSmallIcon(R.drawable.ic_notification)
    .setContentTitle("Um, hi!")
    .setAutoCancel(true)
    .setPriority(NotificationCompat.PRIORITY_HIGH)
    .setFullScreenIntent(pi, true)

val mgr = appContext.getSystemService(NotificationManager::class.java)

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O
    && mgr.getNotificationChannel(CHANNEL_WHATEVER) == null
) {
```

```
        )
      )
    }

    mgr.notify(NOTIF_ID, builder.build())
```

Share Improve this answer Follow

---

1   @SimpleUXApps: If you are trying to start an activity, do not use `PendingIntent.getBroadcast()`. Use `PendingIntent.getActivity()`.
– CommonsWare Sep 7, 2019 at 14:35

1   @steller: I assume that you are referring to this question. **You cannot start an activity from the background in Android 10**, except in very limited cases involving user interaction. This is covered in the documentation. – CommonsWare Sep 28, 2019 at 23:36

3   @steller: "I am making an alarm app and i need to show an alarm screen" -- alarm clock apps have used high-priority notifications with full-screen intents for years. This is covered in the documentation. Your activity will be displayed immediately **if** the user's device is locked, as Android knows that it is safe to show an activity then. And, in other cases, it will not take over the screen from what the user is using it for, as that may cause harm to the user.
– CommonsWare Sep 28, 2019 at 23:44

1   @steller: "im a bit confused by this as my pixel 2 with android 10 is on lock and the intent doesnt fire" -- my guess is that you have not implemented a high-priority notification with a full-screen intent. That is what I was referring to. "Do you have a proper link that shows an example of what your talking about?" -- this is covered in the answer and in the documentation. – CommonsWare Sep 28, 2019 at 23:56

1   @steller: Yes. You might use a pair of actions for that, with associated icons. For example, this activity raises a notification with a full-screen intent and an action. This particular example is not very practical, but it shows how the APIs work. – CommonsWare Sep 29, 2019 at 0:46

▲

10

▼

You can use SYSTEM_ALERT_WINDOW to force launch activity window in android 10, refer to this setting superposition setting:

```
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
<activity
        android:name=".MainActivity"
        android:label="@string/app_name">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
</activity>
    <receiver
        android:name=".OnBootReceiver"
        android:enabled="true"
        android:exported="true"
        android:permission="android.permission.RECEIVE_BOOT_COMPLETED" >
        <intent-filter>
            <action android:name="android.intent.action.BOOT_COMPLETED" />
            <category android:name="android.intent.category.DEFAULT" />
        </intent-filter>
    </receiver>
```

in launched app check permissions:

```
private void RequestPermission() {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
        if (!Settings.canDrawOverlays(this)) {
            Intent intent = new Intent(Settings.ACTION_MANAGE_OVERLAY_PERMISSION,
                    Uri.parse("package:" + this.getPackageName()));
            startActivityForResult(intent,
ACTION_MANAGE_OVERLAY_PERMISSION_REQUEST_CODE);
        } else {
            //Permission Granted-System will work
        }
    }
}
```

you will can user intent as android older versions

```
public class OnBootReceiver extends BroadcastReceiver {
    private static final String TAG = OnBootReceiver.class.getSimpleName();

    @Override
    public void onReceive(Context context, Intent intent) {
        try {
            Intent activity = new Intent(context, MainActivity.class);
            activity.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            context.startActivity(activity);
```

---

answered May 21, 2020 at 15:51

YovanyOso
101   1   2

---

1   this opens the activity even when the app is killed and screen is locked. Thanks. – Noor Hossain Sep 20, 2020 at 9:15

▲

2

▼

🔖

🕘

Android 10's restriction on background activity starts was announced about six months ago. You can read more about it in the documentation.

So you need to have a high-level notification and when the user clicks on the notification your activity will be opened notifications

```java
public class UIExampleReceiver extends BroadcastReceiver {

    public static final String TAG_NOTIFICATION = "NOTIFICATION_MESSAGE";
    public static final String CHANNEL_ID = "channel_1111";
    public static final int NOTIFICATION_ID = 111111;
    private static final String TAG = "Receiver";

    @Override
    public void onReceive(Context context, Intent intent) {
        try {
            // If android 10 or higher
            if (Build.VERSION.SDK_INT > Build.VERSION_CODES.P) {
                startActivityNotification(
                    context,
                    NOTIFICATION_ID,
                    context.getResources().getString(R.string.open_app),
                    context.getResources().getString(R.string.click_app)
                );
            } else {
                // If lower than Android 10, we use the normal method ever.
                Intent activity = new Intent(context, ExampleActivity.class);
                activity.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                context.startActivity(activity);
            }
        } catch (Exception e) {
            Log.d(TAG, e.getMessage() + "");
        }
    }


    // notification method to support opening activities on Android 10
    public static void startActivityNotification (Context context, int
notificationID,
    String title, String message) {

        NotificationManager mNotificationManager =
        (NotificationManager)
        context.getSystemService(Context.NOTIFICATION_SERVICE);
        //Create GPSNotification builder
        NotificationCompat.Builder mBuilder;

        //Initialise ContentIntent
        Intent ContentIntent = new Intent(context, ExampleActivity.class);
        ContentIntent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
        Intent.FLAG_ACTIVITY_CLEAR_TASK);
        PendingIntent ContentPendingIntent = PendingIntent . getActivity (context,
        0,
        ContentIntent,
        PendingIntent.FLAG_UPDATE_CURRENT);

        mBuilder = new NotificationCompat . Builder (context)
            .setSmallIcon(R.drawable.ic_launcher)
            .setContentTitle(title)
            .setContentText(message)
            .setColor(context.getResources().getColor(R.color.colorPrimaryDark))
            .setAutoCancel(true)
            .setContentIntent(ContentPendingIntent)
            .setDefaults(Notification.DEFAULT_LIGHTS | Notification .
DEFAULT_VIBRATE)
            .setCategory(NotificationCompat.CATEGORY_MESSAGE)
            .setPriority(NotificationCompat.PRIORITY_DEFAULT);

        if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.O)
    {
            NotificationChannel mChannel = new NotificationChannel(
                CHANNEL_ID,
                "Activity Opening Notification",
                NotificationManager.IMPORTANCE_HIGH
            );
            mChannel.enableLights(true);
            mChannel.enableVibration(true);
```

```
        }
        Objects.requireNonNull(mNotificationManager).notify(
            TAG_NOTIFICATION, notificationID,
            mBuilder.build()
        );
    }
}
```

Share Improve this answer Follow

edited Mar 23 at 18:40

Jaydip Umaretiya
590 5 27

answered Apr 15, 2020 at 8:57

Huy TRAN
71 6

```
        }
        Objects.requireNonNull(mNotificationManager).notify(
            TAG_NOTIFICATION, notificationID,
            mBuilder.build()
        );
    }
}
```

Join Stack Overflow to find the best answer to your technical question, help others answer theirs.