

# 检查app是否具有推送权限

检查是否有推送权限

```
@RequiresApi(api = Build.VERSION_CODES.KITKAT)
private boolean isNotificationEnabled(Context context) {

    String CHECK_OP_NO_THROW = "checkOpNoThrow";
    String OP_POST_NOTIFICATION = "OP_POST_NOTIFICATION";

    AppOpsManager mAppOps = (AppOpsManager)
context.getSystemService(Context.APP_OPS_SERVICE);
    ApplicationInfo appInfo = context.getApplicationInfo();
    String pkg = context.getApplicationContext().getPackageName();
    int uid = appInfo.uid;

    Class appOpsClass = null;
    /* Context.APP_OPS_MANAGER */
    try {
        appOpsClass = Class.forName(AppOpsManager.class.getName());
        Method checkOpNoThrowMethod = appOpsClass.getMethod(CHECK_OP_NO_THROW,
Integer.TYPE, Integer.TYPE,
        String.class);
        Field opPostNotificationValue =
appOpsClass.getDeclaredField(OP_POST_NOTIFICATION);

        int value = (Integer) opPostNotificationValue.get(Integer.class);
        return ((Integer) checkOpNoThrowMethod.invoke(mAppOps, value, uid, pkg)
== AppOpsManager.MODE_ALLOWED);

    } catch (Exception e) {
        e.printStackTrace();
    }
    return false;
}
```

[跳转到登录页面](#)

```
private void toSetting() {
    Intent localIntent = new Intent();
    localIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
    if (Build.VERSION.SDK_INT >= 9) {
        localIntent.setAction("android.settings.APPLICATION_DETAILS_SETTINGS");

        localIntent.setData(Uri.fromParts("package", getPackageName(), null));
    } else if (Build.VERSION.SDK_INT <= 8) {
        localIntent.setAction(Intent.ACTION_VIEW);
        localIntent.setClassName("com.android.settings",
"com.android.setting.InstalledAppDetails");
        localIntent.putExtra("com.android.settings.ApplicationPkgName",
getPackageName());
    }
}
```

```
startActivity(localIntent);  
}
```