

在 Android 中，如果您的设备已经获得 root 权限，可以将 APK 安装为系统应用。系统应用一般位于 `/system/app` 或 `/system/priv-app` 目录中。下面是将 APK 设置为系统应用的步骤：

1. 将 APK 文件复制到 `/system/priv-app` 或 `/system/app` 目录
 - 使用 ADB 命令或通过应用代码将 APK 复制到系统目录。例如，假设您的 APK 文件名是 `YourApp.apk`：

```
adb root
```

```
adb remount
```

```
adb push YourApp.apk /system/priv-app/YourApp.apk
```

注：大部分新系统中，系统应用通常放在 `/system/priv-app` 目录中。

2. 设置权限
 - 为了确保 APK 文件具有正确的权限，可以运行以下命令：

```
adb shell chmod 644 /system/priv-app/YourApp.apk
```

```
adb shell chown root:root /system/priv-app/YourApp.apk
```

- 644 权限表示文件所有者可以读写，组和其他用户只能读取。

3. 清除 Dalvik/ART 缓存

- 因为安装新的系统应用需要系统加载它，最好清除缓存，以确保应用能够正常识别。

```
adb shell rm -rf /data/dalvik-cache/*
```

4. 重启设备

- 执行重启命令，使更改生效：

```
adb reboot
```

通过代码实现复制

如果您希望通过代码自动完成这些步骤，您可以使用 `su` 权限在应用代码中执行 Shell 命令。

```
fun installAsSystemApp(apkPath: String) {  
    try {  
        val commands = listOf(  
            "mount -o remount,rw /system",  
            "cp $apkPath /system/priv-app/YourApp.apk",  
            "chmod 644 /system/priv-app/YourApp.apk",  
            "chown root:root /system/priv-app/YourApp.apk",  
            "rm -rf /data/dalvik-cache/*",  
            "reboot"  
        )  
    }  
}
```

```
val process = Runtime.getRuntime().exec("su")
val outputStream = process.outputStream
for (command in commands) {
    outputStream.write("$command\n".toByteArray())
    outputStream.flush()
}
outputStream.write("exit\n".toByteArray())
outputStream.flush()
process.waitFor()
} catch (e: Exception) {
    e.printStackTrace()
}
}
```

注意事项

- 修改 /system 目录的写入权限：部分设备上，/system 分区是只读的，必须先重新挂载为可写。可以通过执行 `mount -o remount,rw /system` 来实现。
- 设备安全性：将应用设置为系统应用会增加系统应用层级的权限，适当考虑安全性。