



分类信息

自动抢红包，自动安装原理之 AccessibilityService

> 生活点滴

📅 2016-07-22 19:00:00 | 📄 学习转载 (/article/category/150050) | 👤 转载

(/ARTICLE/CATEGORY/1)

前段时间看别人博客的时候偶然间看到了[Android微信自动回复功能 \(http://www.jianshu.com/p/f67e950d84f7\)](http://www.jianshu.com/p/f67e950d84f7)，最后

> 程序笔记

的效果也很不错，博主在文中提到了AccessibilityService，以前压根没接触过这东西，表示一脸懵逼。也是这个原因

(/ARTICLE/CATEGORY/2)

我去找了AccessibilityService相关的资料好好的看了一遍，发现这个东西真的太NB了，网上对AccessibilityService的

应用还是有不少文章的，但是详细的介绍资料还是比较少，对于刚刚学习这个的同学看完很多资料还是一脸茫然，

于是才有了本文。我相信当你在看完本文之后，再去看前面的那篇文章，我相信会轻松很多。

(/ARTICLE/CATEGORY/3)

AccessibilityService是什么，官网是这样解释的

📄 学习转载

(/ARTICLE/CATEGORY/150050)

Accessibility services are intended to assist users with disabilities in using Android devices and apps. They run in the background and receive callbacks by the system when AccessibilityEvents are fired.

也就是说这是个辅助功能，目的是帮助残疾人去使用Android设备和应用。它在后台运行，可以接收系统的回调。

可见AccessibilityService的出现Google的初衷是帮助残疾人去使用Android设备，但是当你对它足够了解了之后你会发现它的作用不仅仅只是这样。对windows编程有了解的同学肯定知道hook（钩子），AccessibilityService和windows下的hook有那么一点的相似。AccessibilityService可以拦截到系统发出的一些消息（比如窗体状态的改变，通知栏状态的改变，View被点击了等等），当拦截到这些事件我们就可以去做一些我们想做的事儿了~~~

AccessibilityService能做些什么呢？比如自动化测试、自动抢红包、自动安装等等。在带来便利的同时，也还是需要注意的地方：当你开启了辅助功能之后会对你的隐私信息带来一些风险，所以还是需要谨慎的去开启第三方的辅助功能。当然这对于我们开发者而言都不是事，因为我们完全可以自己去开发属于我们的辅助功能。



本文学习目录

1. AccessibilityService 的创建与配置

2. 如何使用 AccessibilityService

3. 一步一步构建一个 apk 自动安装器

> 生活点滴

(/ARTICLE/CATEGORY/1)

1. AccessibilityService 的创建与配置

> 程序笔记

第一步就是自己去创建一个类继承于 AccessibilityService，并实现必须实现的两个方法

(/ARTICLE/CATEGORY/2)

> 程序那点事

```
public class MyAccessibilityService extends AccessibilityService {
```

(/ARTICLE/CATEGORY/3)

```
    public void onAccessibilityEvent(AccessibilityEvent accessibilityEvent) {
```

👍 学习转载

(/ARTICLE/CATEGORY/150050)

```
    @Override
```

```
    public void onInterrupt() {
```

```
    }
```

```
}
```

随后就是在 res/xml 目录下新建一个 xml 文件（文件名随意），后面会对这个文件作详细的介绍。

```
<accessibility-service xmlns:android="http://schemas.android.com/apk/res/android"
    (http://schemas.android.com/apk/res/android)"
    android:accessibilityEventTypes="typeAllMask"
    android:accessibilityFeedbackType="feedbackGeneric"
    android:canRetrieveWindowContent="true"
    android:description="@string/my_accessibility_description"
    android:notificationTimeout="100"
    android:packageNames="com.tencent.mobileqq,com.android.packageinstaller" />
```



最后就是去配置清单文件了



```

<service
    分类信息 android:name=".service.MyAccessibilityService"
    android:enabled="true"
    > 生活点滴 android:exported="true"
        android:label="@string/app_name"
        android:permission="android.permission.BIND_ACCESSIBILITY_SERVICE">
        <intent-filter>
            > 程序笔记 <action android:name="android.accessibilityservice.AccessibilityService" />
        </intent-filter>
        <meta-data
            > 程序那点事 android:name="android.accessibilityservice"
                android:resource="@xml/my_services_config" /> //在meta-data里申明配置信息
    </service>

```

👍 学习转载

别忘记了添加权限

(/ARTICLE/CATEGORY/150050)

```

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

```

到这我们就可以把应用跑到我们的手机上了，然后打开辅助功能，去开启我们刚刚创建的这个辅助功能。如下图：



当然因为什么都没做，开启了也没什么用，下面就去看看怎么使用吧！

2.怎么去使用AccessibilityService

先说说之前创建的那个xml文件中各个属性的含义吧



- accessibilityEventTypes：用来设置响应事件的类型，比如typeAllMask就是响应全部事件，typeNotificationStateChanged就是响应通知状态的改变，如果需要响应多种事件类型可以以 '|' 隔开。
- accessibilityFeedbackType：给用户的反馈方式，比如语音、震动等，这里用处不大。
- canRetrieveWindowContent：是否可以获取活动窗体的内容，这个设置为true才可以取得窗体中的控件和事件源
- description：辅助功能的描述
- notificationTimeout：两个相同类型事件发送到服务的事件间隔，单位毫秒
- packageNames：指定响应某个应用的事件，取值为应用的包名，多个以',' 隔开。没有此属性则表示响应全部应用。这里我填写的是手机qq和系统安装器的包名

> 生活点滴

然后进入到我们的MyAccessibilityService中，定位到onAccessibilityEvent方法编写如下代码
(/ARTICLE/CATEGORY/1)

> 程序笔记

(/ARTICLE/CATEGORY/2)

> 程序那点事

(/ARTICLE/CATEGORY/3)

👍 学习转载

(/ARTICLE/CATEGORY/150050)



```

log("-----");
    int eventType = event.getEventType();//事件类型
    log("packageName:" + event.getPackageName() + "");//响应事件的包名，也就是哪个应用才响应了这个事件

    log("source:" + event.getSource() + "");//事件源信息
    log("source class:" + event.getClassName() + "");//事件源的类名，比如android.widget.TextView
    log("event type(int):" + eventType + "");

    > 生活点滴
        switch (eventType) {
            case AccessibilityEvent.TYPE_NOTIFICATION_STATE_CHANGED:// 通知栏事件
                log("event type:TYPE_NOTIFICATION_STATE_CHANGED");
                break;
            case AccessibilityEvent.TYPE_WINDOW_STATE_CHANGED://窗体状态改变
                log("event type:TYPE_WINDOW_STATE_CHANGED");
                break;
            case AccessibilityEvent.TYPE_VIEW_ACCESSIBILITY_FOCUSED://View获取到焦点
                log("event type:TYPE_VIEW_ACCESSIBILITY_FOCUSED");
                break;
            case AccessibilityEvent.TYPE_GESTURE_DETECTION_START:
                log("event type:TYPE_VIEW_ACCESSIBILITY_FOCUSED");
                break;
            case AccessibilityEvent.TYPE_GESTURE_DETECTION_END:
                log("event type:TYPE_GESTURE_DETECTION_END");
                break;
            case AccessibilityEvent.TYPE_WINDOW_CONTENT_CHANGED:
                log("event type:TYPE_WINDOW_CONTENT_CHANGED");
                break;
            case AccessibilityEvent.TYPE_VIEW_CLICKED:
                log("event type:TYPE_VIEW_CLICKED");
                break;
            case AccessibilityEvent.TYPE_VIEW_TEXT_CHANGED:
                log("event type:TYPE_VIEW_TEXT_CHANGED");
                break;
            case AccessibilityEvent.TYPE_VIEW_SCROLLED:
                log("event type:TYPE_VIEW_SCROLLED");
                break;
            case AccessibilityEvent.TYPE_VIEW_TEXT_SELECTION_CHANGED:
                log("event type:TYPE_VIEW_TEXT_SELECTION_CHANGED");
                break;
        }
    }
}

```



```

    }

    for (CharSequence txt : event.getText()) {
        log("text:" + txt); // 输出当前事件包含的文本信息
    }

    分类信息
    log("-----");

```

> 生活点滴

qq方法就是打印信息用的，是对Log的一个简单封装。

在运行 程序 时，打开我们的手机qq可以看见输出日志如下

```

I/qq: (CATEGORY/1)-----
I/test: packageName:com.tencent.mobileqq
I/test: source:android.view.accessibility.AccessibilityNodeInfo@80007653; boundsInParent:
I/test: source class:com.tencent.mobileqq.activity.SplashActivity
I/test: event type(int):32
I/test: event type:TYPE_WINDOW_STATE_CHANGED
I/test: text:手机QQ
I/test: -----
I/qq: (CATEGORY/1)-----
I/test: packageName:com.tencent.mobileqq
I/test: source:android.view.accessibility.AccessibilityNodeInfo@80007dd5; boundsInParent:
I/test: source class:android.widget.LinearLayout
I/test: event type(int):2048
I/test: event type:TYPE_WINDOW_CONTENT_CHANGED
I/test: -----
I/test: -----
I/test: packageName:com.tencent.mobileqq
I/test: source:android.view.accessibility.AccessibilityNodeInfo@80003308d; boundsInParent:
I/test: source class:android.view.View
I/test: event type(int):2048
I/test: event type:TYPE_WINDOW_CONTENT_CHANGED
I/test: -----
I/test: -----
I/test: packageName:com.tencent.mobileqq
I/test: source:android.view.accessibility.AccessibilityNodeInfo@80007dd5; boundsInParent:
I/test: source class:android.widget.LinearLayout
I/test: event type(int):2048

```

0



可以很清晰的看见当我们打开qq或触发很多的事件，第一个响应的事件是TYPE_WINDOW_STATE_CHANGED,触发此事件的事件源是com.tencent.mobileqq.activity.SplashActivity

接着，当我点击最上面的‘电话’，会得到如下日志：

```
I/test: -----
I/test: packageName:com.tencent.mobileqq
I/test: source:null
I/test: source class:android.view.View
I/test: event type(int):2048
I/test: event type:TYPE_WINDOW_CONTENT_CHANGED
I/test: -----
I/test: -----
I/test: packageName:com.tencent.mobileqq
I/test: source class:android.view.accessibility.AccessibilityNodeInfo@8000c17c; boundsInParen
I/test: event type(int):1
I/test: event type:TYPE_VIEW_CLICKED
I/test: -----
I/test: -----
I/test: packageName:com.tencent.mobileqq
I/test: source class:android.view.accessibility.AccessibilityNodeInfo@8003c309; boundsInParen
I/test: event type(int):2048
I/test: event type:TYPE_WINDOW_CONTENT_CHANGED
I/test: -----
I/test: -----
I/test: packageName:com.tencent.mobileqq
I/test: source:android.view.accessibility.AccessibilityNodeInfo@800402da; boundsInParen
```

我就不一一截图操作的日志了，下来大家可以自行尝试。到这里你用该对onAccessibilityEvent有了进一步的了解。

3.一步一步构建一个apk自动安装器

这一节来个实战的应用：做一个apk的自动安装器，点击apk文件即可开始自动安装。

国内的rom厂商大家都很清楚：百(shang)花(xin)齐(bing)放(kuang)，已经把原生的rom改的面目全非。所以想做一个面对全部手机的apk的自动安装器还是比较麻烦的。我的手机是魅族的，下图是魅族手机apk安装的步骤（其他手机可能会略有不同，下来自己更改不同部分即可，这里主要讲解原理）：





Xhaitun (/main)

15:12

9.31 % 93 40 100



分类信息

> 生活点滴

QQ邮箱

版本: 5.1.3

(/ARTICLE/CATEGORY/1)

> 程序笔记

要安装此应用吗？它将获得以下权限：

(/ARTICLE/CATEGORY/2)

> 电话

程序那点事

读取手机状态和身份

(/ARTICLE/CATEGORY/3)



学习转载

这可能会产生费用
重新设置外拨电话的路径

(/ARTICLE/CATEGORY/150050)

• 相机

拍摄照片和视频

• 麦克风

录音

• 您的位置

大致位置（基于网络）

精确位置（基于GPS和网络）

• 您的社交信息

读取您的通讯录

• 存储

取消

下一步



分类信息

QQ邮箱

生活点滴

版本: 5.1.3

(/ARTICLE/CATEGORY/1)

程序笔记

其他应用的用户界面
(/ARTICLE/CATEGORY/2)


在其他应用之上显示内容

程序那点事

- 影响电池的使用

(/ARTICLE/CATEGORY/3)

控制振动

 [学习转载](#)

● 音频设置

(/ARTICLE/CATEGORY/150050)

- 同步设置

读取同步设置

启用和停用同步

- 系统工具

安装快捷方式

发送持久广播

计算应用存储空间

删除所有应用缓存数据

卸载快捷方式

修改系统设置

取消

安装



可以看见点击安装包之后会弹出一个确认框，点击继续之后会出现下一步，点击了下一步就可以点击安装了。我们要实现自动安装无非就是用程序的方式会对相关按钮的自动点击，难点就在于怎么去找到这些按钮并执行点击操作。

好了，下面我们可以来创建一个自动安装的服务了，步骤和第一节描述的一样将packageNames指定成com.android.packageinstaller就可以了。这里我们先别着急去实现功能（就算你想去实现，也摸不着头脑），我们还是像第二节那样去输出日志信息，查看日志输出信息（对响应事件信息的打印）是使用AccessibilityService的重点，看完日志的输出信息，然后对其分析，才会知道怎么做。

((ARTICLE/CATEGORY/1))
我们来看看在安装apk文件时候输出的部分日志信息：

```

I/test: packageName:com.android.packageinstaller
I/test: source:android.view.accessibility.AccessibilityNodeInfo@8000768f; boundsInParen
I/test: source:android.widget.FrameLayout
I/test: event type(int):2048
I/test: event type:TYPE_WINDOW_CONTENT_CHANGED
I/test: -----
I/test: -----
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:android.view.accessibility.AccessibilityNodeInfo@8000768f; boundsInParen
I/test: source class:android.app.AlertDialog
I/test: event type(int):32
I/test: event type:TYPE_WINDOW_STATE_CHANGED
I/test: text:已安装版本8.3, 降级将会清应用数据, 是否继续?
I/test: text:取消
I/test: text:继续
I/test: -----
I/test: event type:2048
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:null
I/test: source class:android.widget.FrameLayout
I/test: event type(int):2048
I/test: event type:TYPE_WINDOW_CONTENT_CHANGED
I/test: -----

```



Xhaitun (/main)

```

I/test: event type:1
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:null
I/test: source class:android.widget.Button
I/test: event type(int):1
I/test: event type:TYPE_VIEW_CLICKED
I/test: text:继续
I/test: -----
I/test: -----
I/test: event type:32
I/test: -----
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:android.view.accessibility.AccessibilityNodeInfo@800955b4; boundsInParent
I/test: source class:com.android.packageinstaller.PackageInstallerActivity
I/test: event type(int):32
I/test: event type:TYPE_WINDOW_STATE_CHANGED
I/test: text:外部应用安装
I/test: -----
I/test: event type:2048

```

(/ARTICLE/CATEGORY/3)

```

I/test: event type:1
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:android.view.accessibility.AccessibilityNodeInfo@80045d1e; boundsInParent
I/test: source class:android.widget.TextView
I/test: event type(int):1
I/test: event type:TYPE_VIEW_CLICKED
I/test: text:下一步
I/test: -----
I/test: -----
I/test: event type:4096
I/test: -----
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:android.view.accessibility.AccessibilityNodeInfo@8004c63a; boundsInParent
I/test: source class:android.widget.ScrollView
I/test: event type(int):4096
I/test: event type:TYPE_VIEW_SCROLLED
I/test: -----
I/test: -----
I/test: event type:4096
I/test: -----
I/test: -----
I/test: packageName:com.android.packageinstaller
I/test: source:android.view.accessibility.AccessibilityNodeInfo@8004c63a; boundsInParent
I/test: source class:android.widget.ScrollView
I/test: event type(int):4096
I/test: event type:TYPE_VIEW_SCROLLED
I/test: -----

```



从日志信息中可以看出弹出对话框中继续是个Button，接着的下一步是个TextView。到这里我们可以开始编写我们的代码了。

分类信息

> [生活点滴](#)


[\(/ARTICLE/CATEGORY/1\)](#)

> [程序笔记](#)

[\(/ARTICLE/CATEGORY/2\)](#)

> [程序那点事](#)

[\(/ARTICLE/CATEGORY/3\)](#)

 [学习转载](#)

[\(/ARTICLE/CATEGORY/150050\)](#)



```

public class AutoInstallService extends AccessibilityService {

    @Override
    public void onAccessibilityEvent(AccessibilityEvent event) {
        PrintUtils.printEvent(event);
        分类信息
        findAndPerformActionButton("继续");
        findAndPerformActionTextView("下一步");
        > 生活点滴 findAndPerformActionTextView("安装");
    }

    (/ARTICLE/CATEGORY/1)

    > 程序笔记
    private void findAndPerformActionButton(String text) {
        if (getRootInActiveWindow() == null) //取得当前激活窗体的根节点
        return;
        > 程序那点事 通过文字找到当前的节点
        List<AccessibilityNodeInfo> nodes =
        getRootInActiveWindow().findAccessibilityNodeInfosByText(text);
        for (int i = 0; i < nodes.size(); i++) {
            AccessibilityNodeInfo node = nodes.get(i);
            // 执行点击行为
        }
    }

    (/ARTICLE/CATEGORY/150050)
    if (node.getClassName().equals("android.widget.Button") && node.isEnabled()) {
        node.performAction(AccessibilityNodeInfo.ACTION_CLICK);
    }
}

private void findAndPerformActionTextView(String text) {
    if (getRootInActiveWindow() == null)
        return;
    //通过文字找到当前的节点
    List<AccessibilityNodeInfo> nodes =
    getRootInActiveWindow().findAccessibilityNodeInfosByText(text);
    for (int i = 0; i < nodes.size(); i++) {
        AccessibilityNodeInfo node = nodes.get(i);
        // 执行按钮点击行为
        if (node.getClassName().equals("android.widget.TextView") && node.isEnabled()) {
            node.performAction(AccessibilityNodeInfo.ACTION_CLICK);
        }
    }
}

```



```

    }
}

```

例子很简单，代码已经注释的比较详细，我不再对代码作更多的讲解了，有疑问可以留言。

分类信息

到这里你应该对AccessibilityService有了进一步的认识，会发现原来红包助手之类软件的实现的原理也没那么难。是

不是有点自己去做一个红包助手的想法了？现在再去看看文章开始的那篇文章你的收获会更大：[Android微信自动回](http://www.jianshu.com/p/f67e950d84f7)

[复功能 \(http://www.jianshu.com/p/f67e950d84f7\)](http://www.jianshu.com/p/f67e950d84f7)

(/ARTICLE/CATEGORY/1)

程序笔记 补充说明

(/ARTICLE/CATEGORY/2)

AccessibilityService中还有几个常用的方法 onServiceConnected、onInterrupt、onGesture。看名字大致也知道什么

时候会被调用。

(/ARTICLE/CATEGORY/3)

在onServiceConnected中可以去配置AccessibilityService的一些信息，也就是之前在xml文件可以在这里通过代码配

置，不过这里没法配置canRetrieveWindowContent属性，刚开始也被这个坑了很久

(/ARTICLE/CATEGORY/150050)

```

@Override
protected void onServiceConnected() {
    super.onServiceConnected();
    PrintUtils.log("onServiceConnected");
    // 可用代码配置当前Service的信息
    AccessibilityServiceInfo info = new AccessibilityServiceInfo();
    info.packageNames = new String[]{"com.android.packageinstaller",
    "com.tencent.mobileqq", "com.trs.gygdapp"}; //监听过滤的包名
    info.eventTypes = AccessibilityEvent.TYPES_ALL_MASK; //监听哪些行为
    info.feedbackType = AccessibilityServiceInfo.FEEDBACK_SPOKEN; //反馈
    info.notificationTimeout = 100; //通知的时间
    // setServiceInfo(info);
}

```

[官方文档 \(https://developer.android.com/reference/android/accessibilityservice/AccessibilityService.html\)](https://developer.android.com/reference/android/accessibilityservice/AccessibilityService.html)

[本文源码 \(https://github.com/burgessjp/AccessibilityServiceDemo\)](https://github.com/burgessjp/AccessibilityServiceDemo)



文 / _SOLID (简书作者)

原文链接：<http://www.jianshu.com/p/65afab3d1e2a> (<http://www.jianshu.com/p/65afab3d1e2a>)

著作权归作者所有，转载请联系作者获得授权，并标注“简书作者”。

分类信息

转载请注明来源：自动抢红包，自动安装原理之 AccessibilityServic () - Xhaitun (<http://www.xhaitun.com>)

(/ARTICLE/CATEGORY/1)

> 程序笔记

(/ARTICLE/CATEGORY/2)

> 程序那点事

(/ARTICLE/CATEGORY/3)

👍 学习转载

(/ARTICLE/CATEGORY/150050)

Copyright © 2014-2018 xhaitun | Theme By Material Design Lite

粤ICP备14054738号