



Android隐式启动Activity匹配详解：Action，category，data

昵称： tianshiaimili
园龄： 4年8个月
粉丝： 0
关注： 4
[+加关注](#)

<	2019年11月						>
日	一	二	三	四	五	六	
27	28	29	30	31	1	2	
3	4	5	6	7	8	9	
10	11	12	13	14	15	16	
17	18	19	20	21	22	23	
24	25	26	27	28	29	30	
1	2	3	4	5	6	7	

搜索

找找看

谷歌搜索

My Tags

- Linux(3)
- SecureCRT(1)
- wordpress(1)
- 混淆(1)
- 网站(1)
- android(1)
- android intent action(1)
- android 混淆(1)
- Android 特殊字符(1)
- android 隐士启动(1)
- 更多

随笔档案

- 2017年4月(7)
- 2016年10月(1)
- 2016年3月(1)
- 2016年2月(4)
- 2015年12月(5)
- 2015年11月(1)
- 2015年10月(6)
- 2015年9月(6)
- 2015年8月(1)
- 2015年6月(1)
- 2015年5月(2)
- 2015年4月(7)
- 2015年3月(1)

Top Posts

- 1. 关于Android Studio打包混淆以及上传mapping文件(3270)
- 2. android 查看 当前activity(1975)
- 3. 安装Inmp的一些步骤(995)
- 4. android 混淆参数 说明 转的(595)
- 5. LNAP安装(409)

隐式启动Activity的intent到底发给哪个activity，需要进行三个匹配，一个是action，一个是category，一个是data，可以是全部或部分匹配

同样适用于Service和BroadcastReceiver,下面是以Activity为例

MainActivity.java --主Activity

TestActivity.java --需要隐式启动的Activity

(1) 根据Action和Category来进行匹配

```
<activity android:name=".TestActivity"  android:label="TestActivity">

<intent-filter >

    <action android:name="cc.android/myaction.leo"/>

    <category android:name="android.intent.category.DEFAULT"/>

</intent-filter>

</activity>
```

在MainActivity.java里启动它：

```
intent.setAction( "cc.android/myaction.leo");

//不加下面这行也行， 因为intent的这个属性默认值即系Intent.CATEGORY_DEFAULT

intent.addCategory(Intent.CATEGORY_DEFAULT);

startActivity( intent );
```

总结：

a.在某个Activity里用startActivity()方法发送一个intent，这个intent设定了一些条件，比如用方法setAction(),addCategory()设定了两个属性，

发送了这个intent之后，android会去系统里保存的MainManifest.xml清单（假设这个系统存放全部apk清单的文件为MainManifest.xml）里查找符合这两个属性的activity，然后启动它。

查找过程是怎样的呢？

我猜测:在安装某个apk的时候，android系统会把这个apk的清单文件里内容复制一份至系统的某个清单文件里（假如这个系统存放全部apk清单的文件为MainManifest.xml）

当 某个Activity用startActivity(intentOther)方法向系统发送了一个intent(假如为intentOther)，那么 android系统会去查找这个MainManifest.xml里注册的<intent-filter >属性，

查找到符合这个 intentOther 的就启动这个Activity，如果有多个这样的Activity符合条件的话，就跳出一个对话框让用户选择究竟要启动哪一个

上面那个自定义的Action字符串("cc.android/myaction.leo",当然也可以写成这样"cc.android.myaction.leo",同时AndroidManifest.xml里也要写成这样)是系统唯一的，所以系统很容易就能匹配到。

b.任何一个需要隐式启动的Activity都必须要有这项:<category android:name="android.intent.category.DEFAULT"/>

例外情况是： android.intent.category.MAIN和android.intent.category.LAUNCHER的filter中没有必要加入android.intent.category.DEFAULT，当然加入也没有问题

c.假如有两个Activity， 它们的在AndroidManifest.xml里配置如下：

```
<activity android:name="MyActivityOne" android:label="@string/activityOne">
<intent-filter>
<action android:name="hello.leo.liao" />
<action android:name="hello.leo.leo" />
<category android:name="android.intent.category.DEFAULT" />
</intent-filter>
</activity>
```

```
<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">
<intent-filter>
<action android:name="hello.leo.liao" />
<category android:name="android.intent.category.DEFAULT" />
</intent-filter>
</activity>
```

在MainActivity.java里发送一个intent:

```
intent.setAction( "hello.leo.liao");
//不加下面这行也行， 因为intent的这个属性默认值即系Intent.CATEGORY_DEFAULT
intent.addCategory(Intent.CATEGORY_DEFAULT);
startActivity( intent );
```

这样的话， android系统会跳出一个对话框让你选择启动哪一个Activity（MyActivityOne还是MyActivityTwo）

如果把上面的intent.setAction("hello.leo.liao");改为intent.setAction("hello.leo.leo");的话，就自动匹配到MyActivityOne

就是说如果category和action都相同的话，会跳出一个对话框让用户选择要启动哪一个activity;

如果category相同，而action不相同，就可以匹配到相应的activity

d.单单靠添加addCategory属性不能匹配，如：

```
Intent intent = new Intent();
```

```
intent.addCategory(Intent.CATEGORY_DEFAULT);

intent.addCategory("android.intent.category.hello");

startActivity(intent);

<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">

<intent-filter>

<category android:name="android.intent.category.DEFAULT" />

<category android:name="android.intent.category.hello"> </category>

</intent-filter>

</activity>
```

e.当匹配不上任何Activity的话，会发生异常，跳出对话框:很抱歉...某某应用程序意外停止，请重试。

f.Service和BroadcastReceiver 同理

(2) 根据Action和Data匹配

```
<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">

<intent-filter>

    <action android:name="android.intent.action.leo"> </action>

    <category android:name="android.intent.category.DEFAULT"> </category>

    <data android:scheme="x-id"> </data>

</intent-filter>

</activity>
```

//Uri uri = Uri.parse("x-id://www.google.com/getDetails?id=123");//这个也可以

//Uri uri = Uri.parse("x-id");//这个不行

//Uri uri = Uri.parse("x-id://");//这个可以

Uri uri = Uri.parse("x-id:");//这个可以

```
Intent in = new Intent();

in.setAction("android.intent.action.leo");//去掉这行不行,单靠data不能匹配

in.addCategory(Intent.CATEGORY_DEFAULT);//可以去掉这行，因为intent的默认category值即系Intent.CATEGORY_DEFAULT

in.setData(uri);//去掉这行不行

startActivity(in);
```

总结：如果在AndroidManifest.xml里面指定了<data>这行，那么，需要匹配到它的话，在代码里必须要设置intent的data，如上面的in.setData(uri)

Data的语法：

```
<data android:host="string"

    android:mimeType="string"

    android:path="string"

    android:pathPattern="string"
```

```
        android:pathPrefix="string"

        android:port="string"

        android:scheme="string" />
```

Uri的格式:scheme://host:port/path or pathPrefix or pathPattern

如果scheme没有指定，那其它的属性均无效；

如果host没有指定，那么port,path,pathPrefix,pathPattern均无效；

如果在manifest里这样写:<data android:scheme="something"
android:host="project.example.com" />

那么Uri uri = Uri.parse("something://project.example.com"); 才可以匹配

再如：

```
<data android:scheme="something" android:host="project.example.com"
android:port="80"/>
```

等同于这样写：

```
<data android:scheme="something"/>

<data android:host="project.example.com"/>

<data android:port="80"/>
```

那么Uri uri = Uri.parse("something://project.example.com:80"); 才可以匹配

不知为何，下面这个不行：

```
<data android:scheme="content" android:host="com.example.project" android:port="200"
android:path="folder/subfolder/etc"/>
```

Uri uri = Uri.parse("content://com.example.project:200/folder/subfolder/etc")

下面这样也不行

```
<data android:scheme="content" android:host="com.example.project" android:port="200"
android:path="folder"/>
```

Uri uri = Uri.parse("content://com.example.project:200/folder")

可以有多个data,只需匹配其中一个即可

```
<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">

    <intent-filter>

        <action android:name="android.intent.action.leo"></action>

        <category android:name="android.intent.category.DEFAULT"></category>

        <data android:scheme="x-id"/>

        <data android:scheme="something"/>

    </intent-filter>

</activity>

Intent in = new Intent();

in.setAction("android.intent.action.leo");
```

```
        in.addCategory(Intent.CATEGORY_DEFAULT);

        in.setData(Uri.parse("something:")); // 或者用这个亦可 in.setData(Uri.parse("x-id:"));

        startActivity(in);
```

(3) 根据action和data的mimeType属性匹配

```
<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">
<intent-filter>
<action android:name="android.intent.action.VIEW"></action>
<data android:mimeType="vnd.android.cursor.dir/vnd.google.note" />
<category android:name="android.intent.category.DEFAULT"></category>
</intent-filter>
</activity>
```

在java代码里这样写就可以匹配到这个activity:

```
        Intent in = new Intent();

        in.setAction("android.intent.action.VIEW");

        in.addCategory(Intent.CATEGORY_DEFAULT); // 可去掉, 因为Category默认值即系
Intent.CATEGORY_DEFAULT

        in.setType("vnd.android.cursor.dir/vnd.google.note");

        startActivity(in);
```

单靠data的mimeType属性不能匹配, 就算这个mimeType是唯一的也不行(比如
in.setType("leo.android.cursor.dir/vnd.google.leo");), 需要有一个action配合

可以有多个mimeType, 在java代码里只需匹配其中一个即可:

```
<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">
<intent-filter>
    <intent-filter>
<action android:name="android.intent.action.VIEW"></action>
<data android:mimeType="leo.android.cursor.dir/vnd.google.leo" />
<data android:mimeType="leo.android.cursor.dir/vnd.google.liao" />
<category android:name="android.intent.category.DEFAULT"></category>
</intent-filter>
</activity>
```

这样可以启动MyActivityTwo这个Activity:

```
Intent in = new Intent();

        in.setAction("android.intent.action.VIEW");

        in.addCategory(Intent.CATEGORY_DEFAULT); // 可去掉, 因为Category默认值即系
Intent.CATEGORY_DEFAULT

        in.setType("leo.android.cursor.dir/vnd.google.liao");

        startActivity(in);
```

或者这样也可以启动MyActivityTwo这个Activity:

```
Intent in = new Intent();
```



```
in.setAction("android.intent.action.VIEW");

in.addCategory(Intent.CATEGORY_DEFAULT);//可去掉,因为Category默认值即系
Intent.CATEGORY_DEFAULT

in.setType("leo.android.cursor.dir/vnd.google.leo");

startActivity(in);
```

scheme和mimeType只能有其中一个,下面这样通不过

AndroidManifest.xml里:

```
<data android:scheme="something" android:host="project.example.com"
android:port="80"
```

```
android:mimeType="leo.android.cursor.dir/vnd.google.leo" />
```

或

```
<data android:scheme="something" android:host="project.example.com"
android:port="80" />
```

```
<data android:mimeType="leo.android.cursor.dir/vnd.google.leo" />
```

java代码里:

匹配不上:

```
Intent in = new Intent();

in.setAction("android.intent.action.VIEW");

Uri uri = Uri.parse("something://project.example.com:80");

in.setData(uri);

in.setType("leo.android.cursor.dir/vnd.google.leo");

startActivity(in);
```

这样还是匹配不上:

```
Intent in = new Intent();

in.setAction("android.intent.action.VIEW");

//      Uri uri = Uri.parse("something://project.example.com:80");
//      in.setData(uri);

in.setType("leo.android.cursor.dir/vnd.google.leo");

startActivity(in);
```

这样还是匹配不上:

```
Intent in = new Intent();

in.setAction("android.intent.action.VIEW");

Uri uri = Uri.parse("something://project.example.com:80");

in.setData(uri);

//      in.setType("leo.android.cursor.dir/vnd.google.leo");

startActivity(in);
```

(4) 一个Activity里可以有多对<intent-filter></intent-filter> 只要匹配其中一对, 即可启动这个Activity

```
<activity android:name=".MyActivityTwo" android:label="@string/activityTwo">
<intent-filter>
<action android:name="android.intent.action.VIEW"></action>
<data android:scheme="something" android:host="project.example.com"
android:port="80" />
<category android:name="android.intent.category.DEFAULT"></category>
</intent-filter>
<intent-filter>
<action android:name="android.intent.action.VIEW"></action>
<data android:mimeType="leo.android.cursor.dir/vnd.google.leo" />
<category android:name="android.intent.category.DEFAULT"></category>
</intent-filter>
<intent-filter>
<action android:name="hello.hi.liao"></action>
<category android:name="android.intent.category.DEFAULT"></category>
</intent-filter>
</activity>
```

java代码里:

匹配第一对<intent-filter> 可以启动MyActivityTwo这个Activity:

```
Intent in = new Intent();

        in.setAction("android.intent.action.VIEW");

        in.addCategory(Intent.CATEGORY_DEFAULT);//可去掉,因为Category默认值即系
Intent.CATEGORY_DEFAULT

        Uri uri = Uri.parse("something://project.example.com:80");

        in.setData(uri);

        startActivity(in);
```

匹配第二对<intent-filter> 也可以启动MyActivityTwo这个Activity:

```
Intent in = new Intent();

        in.addCategory(Intent.CATEGORY_DEFAULT);//可去掉,因为Category默认值即系
Intent.CATEGORY_DEFAULT

        in.setAction("android.intent.action.VIEW");

        in.setType("leo.android.cursor.dir/vnd.google.leo");

        startActivity(in);
```

匹配第三对<intent-filter> 也可以启动MyActivityTwo这个Activity:

```
Intent in = new Intent();

        in.setAction("hello.hi.liao");

        in.addCategory(Intent.CATEGORY_DEFAULT);//可去掉,因为Category默认值即系
Intent.CATEGORY_DEFAULT

        startActivity(in);
```

全部总结：

1. `<action/>`包含在 `<intent-filter></intent-filter>` 标签对里，而且是必不可少的！不管以哪一种方式来匹配，都不可缺少这个`<action/>` ,可以有多个，至少要有一个。

如有多个的，话只需要匹配其中一个即可找到这个activity

`<action>`里的属性值大多数是在Intent里定义的，比如`<action android:name="android.intent.action.VIEW"/>`里的属性值就等于 `Intent.ACTION_VIEW`, 在这个Intent类里以ACTION开头定义的常量都是。当然，也可以自定义。

2. 任何一个需要隐式启动的Activity都必须要有这项：`<category android:name="android.intent.category.DEFAULT"/>`

例外情况是：`android.intent.category.MAIN`和`android.intent.category.LAUNCHER`的filter中没有必要加入`android.intent.category.DEFAULT`，当然加入也没有问题

`<category>` 里的属性值大多数是在Intent里定义的，比如 `<category android:name="android.intent.category.DEFAULT"/>`里的属性值就等于 `Intent.CATEGORY_DEFAULT`,

在这个Intent类里以CATEGORY开头定义的常量都是。当然，也可以自定义。

3. 一个Activity里可以有多对`<intent-filter></intent-filter>` 只要匹配其中一对，即可启动这个Activity

4. 在`<intent-filter></intent-filter>`里可以有多个`<data android:mimeType="xxxx"/>`，只需匹配其中一个即可.注意:不可以同时出现第5点的标签对，即下面这条。

5. 在`<intent-filter></intent-filter>`里可以有多个`<data android:scheme="xxxx" android:host="yyyy" android:port="uuu"/>`，只需匹配其中一个即可。

语法：

```
<data android:host="string"
      android:mimeType="string"
      android:path="string"
      android:pathPattern="string"
      android:pathPrefix="string"
      android:port="string"
      android:scheme="string" />
```

可以分开写，如：

```
<data android:scheme="something" android:host="project.example.com"
      android:port="80"/>
```

等同于这样写：

```
<data android:scheme="something"/>
<data android:host="project.example.com"/>
```


<data android:port="80"/>

在java代码里,Uri的格式:scheme://host:port/path or pathPrefix or pathPattern

注意:不可以同时出现第4点的标签对，即上面那条。

6.在<intent-filter></intent-filter>里可以有多个<action android:name="xxxx"> ,只需匹配其中一个即可。

7.当匹配不上任何Activity的话，会发生异常，跳出对话框:很抱歉...某某应用程序意外停止，请重试。

8. 上面所说的全部适用于Service和BroadcastReceiver，只需把<activity ...></activity>换成<service ...></service>或<receiver ...></receiver>即可。

9.刚参考了一下[packages\apps\HTMLViewer\AndroidManifest.xml](#),第4和第5条应该是不冲突才对，但是实际测试中却是冲突，暂时未到找原因。匹配方式请看:[用于打开HTML文件的intent](#)

在被启动的Activity(本例为MyActivityTwo)里接收数据:

```
Intent intent = getIntent();
```

```
String intentCategories = intent.getCategories()
```

```
String intentType = intent.getType();
```

```
Uri uri = intent.getData();
```

```
String uriScheme = uri.getScheme();
```

```
String uriPath = uri.getPath();
```

```
String uriHost = uri.getHost();
```

```
String uriEncodedPath = uri.getEncodedPath();
```

请参考:packages\apps\HTMLViewer\src\com\android\htmlviewer\HTMLViewerActivity.java

标签: [android 隐士启动](#)

标签: [android 隐士启动](#)

好文要顶

关注我

收藏该文

[tianshiaimili](#)

[关注 - 4](#)

[粉丝 - 0](#)

[+加关注](#)

0

推荐

0

反对

« 上一篇: [Android NotificationManager 和Notification的使用总结](#) [转](#)

» 下一篇: [android 混淆参数 说明](#) [转的](#)

posted @ 2015-10-14 00:12 tianshiaimili 阅读(141) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)， [访问](#) [网站首页](#)。

【推荐】超50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【活动】京东云服务器_云主机低于1折，低价高性能产品备战双11

【培训】马士兵老师强势回归！Java线下课程全免费，双十一大促！

【推荐】天翼云双十一翼降到底，云主机11.11元起，抽奖送大礼

- 【推荐】 流程自动化专家UiBot，体系化教程成就高薪RPA工程师
- 【福利】 个推四大热门移动开发SDK全部免费用一年，限时抢！
- 【推荐】 阿里云双11冰点钜惠，热门产品低至一折等你来抢！

相关博文：

- 关于Android隐式启动Activity
 - android Action中的data属性
 - android intent-filter
 - 隐式Intent
 - Android隐式启动匹配： action,category,data
- » 更多推荐...

最新 IT 新闻：

- 阿里的双 11，除了破纪录，还是一场 AI 秀
 - 人类制造的那些垃圾何都到哪里去了？
 - 欧空局正尝试在国际空间站培育血管
 - 双十一，程序员可太难了
 - 2799元起！小米CC9 Pro正式开售： 1亿像素加持DXOMark总分第一
- » 更多新闻...

