

[置顶] android 自定义权限 permission

2011-06-17 22:01 1123 人阅读 [评论\(1\)](#) [收藏](#) [举报](#)

Android 支持为应用程序自定义权限。如果希望自己的程序组件(如 Activity, BroadcastReceiver)等被任意用户随意启动执行。要使用自定义权限, 首先在 AndroidManifest.xml 文件中声明它们, 定义了权限之后, 可以将它作为组件定义的一部分进行引用。

下面列举一个 demo, 详述自定义权限的过程:
首先创建一个工程 CustomMadePermisson:

主 Activity(启动 Activity):

```
import android.app.Activity;
import android.os.Bundle;
import android.view.ViewGroup.LayoutParams;
import android.widget.LinearLayout;
import android.widget.TextView;

public class MainActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        LinearLayout layout = new LinearLayout(this);
        layout.setLayoutParams(new LayoutParams(LayoutParams.FILL_PARENT,
NT,
        LayoutParams.WRAP_CONTENT));
        layout.setOrientation(LinearLayout.HORIZONTAL);
        TextView textView = new TextView(this);
        textView.setText("欢迎来到徐志奇的博客!");
        layout.addView(textView);
        setContentView(layout);
    }
}
```

新建另外一个 Activity，名称为 Another，该 Activity 只是一个空 Activity，用来验证添加启动权限后的启动效果：

```
import android.app.Activity;
import android.os.Bundle;
import android.widget.Toast;

public class Another extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        Toast.makeText(this, "欢迎来到新的界面", Toast.LENGTH_LONG).show();
    }
}
```

添加权限：

双击 AndroidManifest.xml 文件，打开 permissions 选项卡，在 Permissions 窗口中单击 Add 按钮，选择 Permission，然后单击 ok 按钮，添加一个新的 Permission：

权限的特性：

android:name

权限的名称，必填属性，通常应该遵循 Android 的命名方案 (*.permission.*) 但非必须。

android:protectionLevel

定义与权限相关的保护级别，必填属性。必须选择一下四项之一：normal、dangerous、signature、signatureOrSystem。

android:permissionGroup

非必填属性，可以将权限放在一个组中，但对于自定义权限，尽量不要设置此属性了。

android:label

非必填属性，含义你应该明白。

android:description

非必填属性，含义你应该明白。

android:icon

非必填属性，含义你应该明白。

现在已经定义好了权限，接下来就是设置启动 Activity 的权限了。要在活动上设置权限，可以将 android:permission 特性添加到 AndroidManifest.xml 文件中：

AndroidManifest.xml 文件：

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.xzq.custompermission" android:versionCode="1"
    android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />
    <permission android:protectionLevel="normal"
        android:name="com.xzq.permission.zhongnantest"></permission>

    <application android:icon="@drawable/icon" android:label="@string/
/app_name">
        <activity android:name=".MainActivity" android:label="@string/
app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHE
R" />
            </intent-filter>
        </activity>
        <activity android:name=".Another" android:label="@string/app_
name"
            android:permission="com.xzq.permission.zhongnantest">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHE
R" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

新建测试工程 Clint:

在该工程中创建一个 Activity，用来测试上面添加了启动权限的 Activity 的启动情况。

```
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        ((Button) findViewById(R.id.button1))
            .setOnClickListener(new View.OnClickListener() {

                @Override
                public void onClick(View v) {
                    Intent intent = new Intent();
                    intent.setClassName("com.xzq.custompermission",
                        "com.xzq.custompermission.Another");
                    startActivity(intent);
                }
            });
    }
}
```

界面布局:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/andro
id"
```

```

        android:orientation="vertical" android:layout_width="fill_parent"
        android:layout_height="fill_parent">
        <TextView android:layout_width="fill_parent"
            android:layout_height="wrap_content" android:text="@string/hello" />
        <Button android:text="启动新Activity" android:id="@+id/button1"
            android:layout_width="wrap_content" android:layout_height="wrap_content"></Button>
    </LinearLayout>

```

AndroidManifest.xml 文件:

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.xzq.client" android:versionCode="1" android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />
    <uses-permission android:name="com.xzq.permission.zhongnantest" />
    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <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>
    </application>
</manifest>

```

重点是在该文件中添加了相应的权限，才可以启动上面的界面。

自定义权限不仅适用于活动，实际上，也可以同样应用于 Android 中的其他类型的组件。