北京基调网络股份有限公司

移动 App 监测 SDK 部署说明书

Android SDK Maven 部署文档

Meng'yang Cheng

2016/5/5

目录

听云 APP 部署文档 2

1、设置应用名称 2

2、使用 MAVEN 构建 2

3、嵌码验证 4

4、性能数据采集 4

5、CRASH 反混淆 5

5.1、查找 Mapping File 文件 5

6、配置 TINGYUN.PROPERTIES 文件 6

6.1、新建配置文件 6

6.2、配置 Mapping File 自动上传 6

6.3、配置 OkHttp 控制开关 8

6.4、配置 WebView 控制开关 8

7、配置用户自定义附加信息 8

附录 9

听云 APP 支持协议类库 9

听云 TCP 支持设备版本 9

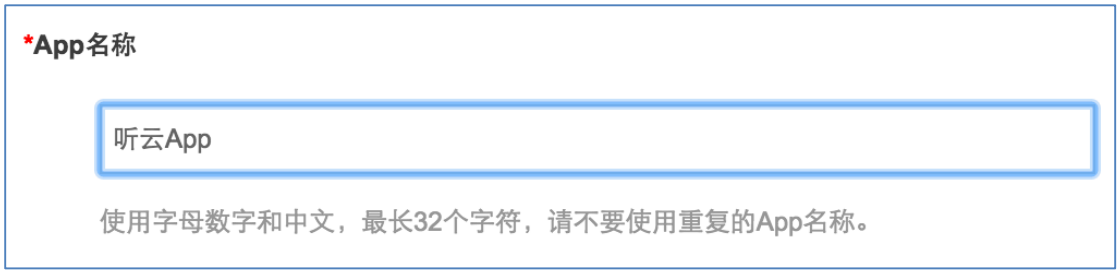
听云 App 部署文档

1、设置应用名称

请为需要监测的 App 设置一个易于识别的应用名称，该 App 名称将出现在 App 列表和相关的报表中来代表对应的 App。

App 名称可以使用字母数字和中文，最长 32 个字符。设置 App 名称时，尽量不要设置重复的 App 名称，以免给自己造成不必要的麻烦。

您的 App 名称可以在设置选项中随时修改



2、使用 Maven 构建

（1）将以下的配置内容合并到你的 pom.xml 文件中：

```
<dependencies>
    <dependency>
        <groupId>com.networkbench.newlens.agent.android</groupId>
        <artifactId>nbs.newlens.agent</artifactId>
        <version>TingYun_Version</version>
    </dependency>
</dependencies>

<build>
 <plugins>
    <plugin>
        <groupId>com.networkbench.newlens.agent.android</groupId>
        <artifactId>agent-maven-plugin</artifactId>
    <version> TingYun_Version </version>
    <executions>
      <execution>
```

```
            <goals>
                <goal>instrument</goal>
            </goals>
        </execution>
    </executions>
  </plugin>
</build>
```

注：TingYun_Version 为当前听云 App 版本号，可登陆官网查询最新版本号：http://www.tingyun.com/。如：2.3.0。

（2）设置应用程序权限

请在待监测的 App 工程的 AndroidMainfest.xml 文件中增加以下的权限：

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_LOGS" />
<uses-permission android:name="android.permission.INTERNET" />
```

（3）插入启动 NBSAgent 的代码

在待监测 App 的主 Activity（Main Activity）源文件中导入听云的 Android 探针类：

```
import com.networkbench.agent.impl.NBSAppAgent;
```

在 onCreate()方法中（如果没有请新增 onCreate()方法）增加以下初始化听云 App 探针的调用：

```
NBSAppAgent.setLicenseKey("AppKey
").withLocationServiceEnabled(true).start(this.getApplicationContext());
```

注：如项目使用了腾讯 X5 框架，可调用 setX5Enable()方法

```
NBSAppAgent.setLicenseKey("AppKey
").withLocationServiceEnabled(true).setX5Enable(true).start(getApplicationContext());
```

若不需要采集地理位置信息（报表中将没有区域维度的数据）请使用以下代码：

```
NBSAppAgent.setLicenseKey("AppKey ").start(this.getApplicationContext());
```

其中“AppKey”从听云 App 系统中生成。

（4）设置混淆器

如果待监测的 App 使用 ProGuard 工具来优化发布的 apk 包，为了保证听云 App 监测 SDK 可以正常使用，请在所属项目的 proguard.cfg 文件中加入以下的内容：

```
# ProGuard configurations for NetworkBench Lens
-keep class com.networkbench.** { *; }
-dontwarn com.networkbench.**
-keepattributes Exceptions, Signature, InnerClasses
# End NetworkBench Lens
```

如果您希望保留行号信息，建议您在 proguard.cfg 中添加如下代码：

```
-keepattributes SourceFile,LineNumberTable
```

（5）clean 项目工程

建议在 Terminal 下 Clean 项目工程，以确保听云 App 配置生效

```
mvn clean
```

（6）运行 apk

打包编译后，在模拟器或 Android 设备上运行待监测的 App，并进行适当的网络访问。

3、嵌码验证

运行嵌码完成后的 APK，连接电脑使用 LogCat 过滤 TAG = NBSAgent 查看 log 日志

出现如下 log 表示嵌码成功：

```
NBSAgent start
NBSAgent enabled
NBSAgent V “TingYun_Version”
connect success
```

4、性能数据采集

若以上的步骤操作正确，听云 App Android SDK 探针就已经成功部署到待测 App。保持 App 运行，并访问网络，静待几分钟即可在听云报表看到嵌码 App 的性能数据。

5、Crash 反混淆

为了更好的帮助用户解析崩溃信息,我们提供了反混淆机制来帮助用户快速精准的定位崩溃发生位置。当您使用反混淆机制时,需要您上传当前版 本的 Mapping File 到听云服务器。

目前,我们提供了两种方式上传 Mapping File,您可以选择任意一种方式上传您的 Mapping File 此过程不会干扰正常用户代码和打包编译。

5.1、查找 Mapping File 文件

进入嵌码项目根目录，搜索 Mapping.txt。

5.1.1、报表上传对应版本的 Mapping File



i. 登陆听云报表，在 App 概览中选择嵌对应的 App 名称，进入 App 后选择左侧菜单栏中的“崩溃”按钮：



ii. 在右侧“崩溃历史记录列表”中选择并点击“dSYM / Mapping 文件管理”



iii. 在弹出窗口中选择上传平台类型，如：“Android”。选择对应的应用版本号，并点击右侧上传按钮，手动上传 Mapping File

6、配置 tingyun.properties 文件

6.1、新建配置文件

进入嵌码项目根目录，新建一个 tingyun.properties 配置文件

6.2、配置 Mapping File 自动上传

将下列配置项添加到到 tingyun.properties 配置文件：

```
authKey=听云 API 账号授权 Key，由报表系统生成
appKey=听云 App 授权 Key
mapping_file_auto_upload=上传 Mapping 开关
```

1) 配置 AuthKey 授权码



i. 登陆报表，选择用户名称，点击修改授权码



ii. 复制“当前授权码”并黏贴到配置文件下

2) 配置 AppKey



i. 登陆报表，在 App 概览中选择嵌对应的 App 名称，如：Android_SDK。点击左侧配置按钮，选择“修改设置”



ii. 进入“修改配置”后，找到“安装 SDK”选择“App Key”复制到配置文件中

3) 配置 mapping_file_auto_upload 上传开关

如果您需要自动上传 Mapping File，请在配置文件中打开 Mapping File 控制开关：

```
mapping_file_auto_upload=true
```

注意：改开关只有在启用混淆器时才会生效，通过主机上传 Mapping File 到听云服务器，您可以在正式发布或需要时打开 Mapping File 上传开关，如应用未开启混淆器，听云 SDK 将不会启用配置文件

6.3、配置 OkHttp 控制开关

将下列配置项添加到到 tingyun.properties 配置文件：

```
okhttp3.0=true
```

注：听云默认对 OkHttp2.0 嵌码，如果您使用 Okhttp2.0 请忽略该设置，若使用的是 OkHttp1.0 请设置 okhttp2.0=false 即可

6.4、配置 WebView 控制开关

将下列配置项添加到到 tingyun.properties 配置文件：

```
webview=true
```

注：听云 WebView 需使用 setWebViewClient 方法，如未实现该方法，请添加如下代码

```
webview.setWebViewClient(new WebViewClient() {});
```

7、配置用户自定义附加信息

听云 App 提供用了户自定义附加信息接口，用户可以自定义配置附加信息。当应用发生 Crash 时上传到报表服务器。用户可以再初始化之后任意位置配置该接口。UserCrashMessage 最多可添加 10 条附加信息，每条附加信息最大支持 100 个字节。

用户自定义附加接口如下：

```
NBSAppAgent.setUserCrashMessage(String key,String value)
```

附录

听云 App 支持协议类库

- HttpURLConnection
- Android HttpClient 4.0
- Apache HttpClient (> 4.0)
- Volley+OkHttpClient
- OkHttp 2.0 (> 2.1)
- OkHttp 1.0
- Retrofit (> 1.9)

听云 TCP 支持设备版本

- 目前只支持 Android 4.0