

符号表工具Android版-使用指南

Bugly文档资料

符号表工具Android版-使用指南

1. 介绍

- 1.1 环境要求
- 1.2 符号表提取要求
- 1.3 配置文件
- 1.3 上传功能

2. 用法

- 2.1 工具使用方法
- 2.2 支持的选项
- 2.3 举一个例子

3. Debug SO文件

- 3.1 什么是Debug SO文件？
- 3.2 如何定位Debug SO文件？
 - 3.2.1 IDE: Eclipse
 - 3.2.2 IDE: Android Studio

1. 介绍

为了能快速并准确地定位用户App发生**Crash的代码位置**，Bugly使用**符号表文件**对App发生Crash的程序**堆栈**进行**解析**和**还原**。

举一个实例：

还原前堆栈

```
#00 pc 0021cdf4 /lib/libgame.so
#01 pc 002abe36 /lib/libgame.so
#02 pc 003aebec /lib/libgame.so
```

还原后堆栈

```
#00 pc 0021cdf4 _ZNK14CAAnimationNode13getCurEquipIdEv (CAAnimationNode.h:51)
#01 pc 0021cdf4 _ZNSt6vectorI15NDKCallbackNodeSaIS0_EE5beginEv (NDKHelper.cpp:312)
#02 pc 0021cdf4 _ZNK6b2Vec26LengthEv (b2Math.h:121)
```

而符号表工具，正是Bugly提供给开发者提取符号表文件（.symbol）的工具。另外，Bugly提供了自动上传符号表文件的方法，请参考《[Bugly Android 符号表配置](#)》，建议使用自动上传的方式。

1.1 环境要求

符号表工具的运行需要[Java运行环境](#)（ Java SE Runtime Environment ），JRE或JDK版本需要 ≥ 1.6 。

1.2 符号表提取要求

提取符号表需要[符号表工具](#)和**Debug SO文件**（ 具有调试信息的SO文件，可参考下文的第三部分：**3. Debug SO文件** ）。

1.3 配置文件

Bugly Android符号表工具2.5.0及以上版本增加了配置文件的解析功能，工具包中提供了一个与工具JAR同目录的默认配置文件（ settings.txt ）。

可以通过配置文件设置以下信息：

- Debug：调试模式开关（ 打印更多Log ）
- Upload：上传开关
- ID：Bugly平台的App ID
- Key：Bugly平台的App key

1.3 上传功能

Bugly Android符号表工具2.5.0及以上版本增加了上传功能，并支持Mapping文件的上传。

使用上传功能时，需要指定以下信息：

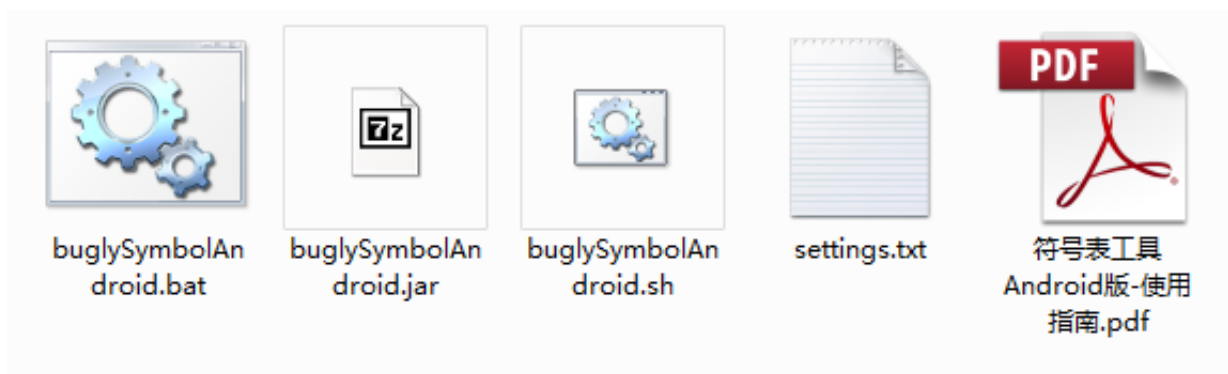
- App ID（ 可通过配置文件指定 ）
- App key（ 可通过配置文件指定 ）
- App版本
- App包名

目前脚本不支持以上信息的指定，因此需要通过直接执行JAR包来使用上传功能。

2. 用法

Android版符号表工具支持Windows、Linux和Mac三个平台，同时提供了JAR包、各平台脚本和符号表配置文件：

- 符号表工具JAR包（ buglySymbolAndroid.jar ）
- Windows的脚本（ buglySymbolAndroid.bat ）
- Shell脚本（ buglySymbolAndroid.sh ）
- 默认符号表配置文件（ settings.txt ）



使用脚本时，请保证脚本和jar包在同个目录下！

2.1 工具使用方法

```
java -jar <JAR包> <选项>

<Bat脚本> <选项>

<Shell脚本> <选项>
```

2.2 支持的选项

选项	说明
-i	指定文件路径，可指定目录（必选）
-o	输出的符号表zip文件的路径，必须是zip文件
-d	调试模式开关（默认关闭）
-s	指定配置文件（默认读取JAR目录下的“settings.txt”文件）
-u	上传开关
-id	App ID
-key	App key
-package	App包名
-version	App版本

2.3 举一个例子

注意事项

不要直接复制例子中的命令运行，需要根据自己的具体情况更改下命令。

环境和用户信息

- 系统：**Windows**
- Android工程目录：**E:\Projects\Demo**
- 符号表工具（已解压）所在目录：**D:\Downloads\buglySymbolAndroid**
- Debug SO所在目录：**E:\Projects\Demo\obj**
- APP ID：**900012345**
- APP key：**abcdefghijk**
- APP包名：**com.batman.demo**
- APP版本：**2.3.1**

生成符号表文件

使用符号表工具的JAR包生成符号表文件的命令如下：

```
cd D:\Downloads\buglySymbolAndroid

java -jar buglySymbolAndroid.jar -i E:\Projects\Demo\obj
```

生成的符号表文与库文件（test.app.dSYM）位于同级目录：/Users/batman/Desktop/

生成符号表文件并自动上传

使用符号表工具的JAR包生成符号表文件，并自动上传的命令如下：

```
cd D:\Downloads\buglySymbolAndroid

java -jar buglySymbolAndroid.jar -i E:\Projects\Demo\obj -u -id 900
012345 -key abcdefghijk -package com.batman.demo -version 2.3.1
```

3. Debug SO文件

3.1 什么是Debug SO文件？

Android 平台中，目标文件对应的是SO文件。Debug SO文件是指具有调试信息的SO文件。

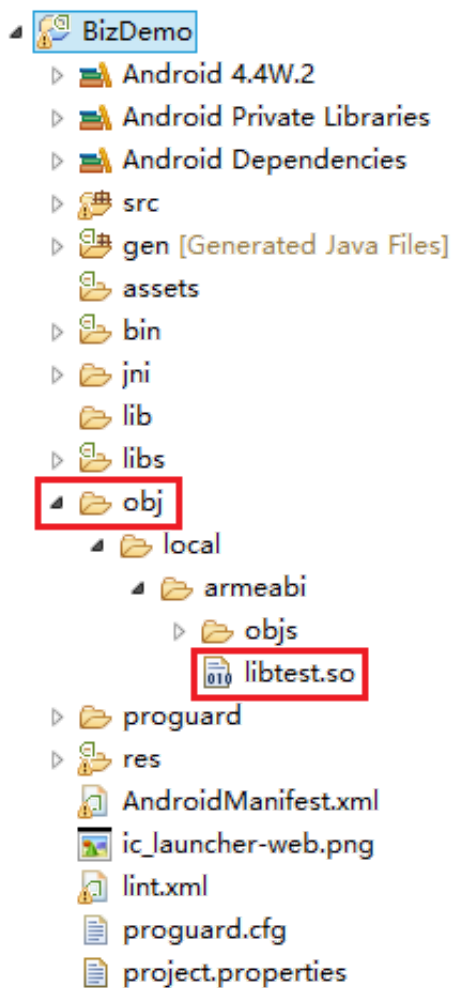
3.2 如何定位Debug SO文件？

3.2.1 IDE: Eclipse

IDE如果使用Eclipse+NDK，默认情况下，Debug SO文件将位于：

<项目文件夹>/obj/local/<架构(Architecture)>/

如下图所示：



3.2.2 IDE: Android Studio

IDE如果使用Android Studio+NDK，默认情况下，**Debug**编译的Debug SO文件将位于：

`<项目文件夹>/<Module>/build/intermediates/ndk/debug/obj/local<架构>/`

而**Release**编译的Debug SO文件将位于：

`<项目文件夹>/<Module>/build/intermediates/ndk/release/obj/local<架构>/`

如下图所示：

