|  |
| --- |
| 百度 |
| 百度移动统计iOS crash日志解析指南 |
|  |
|  |
| **MTJ** |
| **2015/2/28** |

|  |
| --- |
|  |

**修改记录**

|  |  |
| --- | --- |
| **更新时间** | **变更内容** |
| 2015.02.28 | 全文 |
|  |  |

目录

第一章 简介 4

第二章 阅读对象 4

第三章 crash日志格式 4

第四章 解析说明与材料准备 5

一 解析说明 5

二 材料准备 5

第五章 解析方法 5

第一节 使用symbol address解析（方法一） 5

一. 获取app启动时候加载的起始地址。 5

二. 计算symbol address 6

三. 解析symbol address 6

第二节 使用load address相对偏移解析（方法二） 7

一． 计算load address相对偏移 7

二． 使用load address解析 7

第六章 联系我们 7

# 简介

本文档指在帮助使用百度移动统计iOS SDK的用户，解析通过SDK获取到的crash日志内容，以便定位发生crash的代码位置。

未使用百度移动统计iOS SDK的读者可以下载SDK使用。SDK下载地址：

<http://mtj.baidu.com/web/welcome/sdk>，包的详细介绍见第六章。

# 阅读对象

本文档面向所有使用百度移动统计iOS SDK 的开发人员、测试人员。

# crash日志格式

百度移动统计iOS SDK获取的crash日志不是完整的系统生成日志，其内容主要包括异常类型、异常原因、发生Crash的线程的Crash调用栈等。如下所示

Hi

使用app时发生了崩溃, 详细错误信息如下:

\*\*\* Terminating app due to uncaught exception 'NSRangeException', reason: '\*\*\* -[\_\_NSArrayM objectAtIndex:]: index 3 beyond bounds [0 .. 1]'

详细错误信息:

(

0 CoreFoundation 0x0000000186ebe5b8 + 160

1 libobjc.A.dylib 0x00000001975c40e4 objc\_exception\_throw + 60

2 CoreFoundation 0x0000000186da362c + 0

3 BaiduStatApp 0x000000010009e824 BaiduStatApp + 288804

4 BaiduStatApp 0x00000001000aca04 BaiduStatApp + 346628

5 BaiduStatApp 0x00000001000a56e8 BaiduStatApp + 317160

6 BaiduStatApp 0x00000001000bf754 BaiduStatApp + 423764

7 Foundation 0x0000000187d91d90 + 372

8 CoreFoundation 0x0000000186e769ec + 24

9 CoreFoundation 0x0000000186e75c90 + 264

10 CoreFoundation 0x0000000186e73d40 + 712

11 CoreFoundation 0x0000000186da10a4 CFRunLoopRunSpecific + 396

12 GraphicsServices 0x000000018ff435a4 GSEventRunModal + 168

13 UIKit 0x000000018b6d2aa4 UIApplicationMain + 1488

14 BaiduStatApp 0x000000010007f318 BaiduStatApp + 160536

15 libdyld.dylib 0x0000000197c32a08 + 4

)

# 解析说明与材料准备

## 一 解析说明

1.由于日志格式不是完整的iOS crash日志，所以无法使用Xcode自带的symbolicatecrash工具进行整体日志解析。这里我们使用系统命令进行逐行解析。

2.对于系统库所对应的函数定位（例如Foundation、UIKit库等）这里不做解析，解析的是开发者自定义的项目工程中的代码位置（例如上图中BaiduStatApp），也是对于开发者定位函数位置最有价值的信息。

## 二 材料准备

所需材料：发布时的.app文件。

若提交时候使用的是.xcarchive文件，则可以通过右键>显示包内容，然后通过路径.xcarchive/Products/Applications/xxx.app找到。

# 解析方法

通过如下步骤进行解析：

## 第一节 使用symbol address解析（方法一）

### 一. 获取app启动时候加载的起始地址。

起始地址：即使每次iOS app启动都会加载(main module)主模块在不同的内存地址（大多数情况下32bit框架对应的地址是0x4000、64bit框架对应的地址为0x0000000100000000）。

获取此地址的方法：

1.打开终端，cd进入.app文件所在目录。

2.输入指令

otool -arch arm64 -l xxx.app/xxx | grep -B 1 -A 10 "LC\_SEGM" | grep -B 3 -A 8 "\_\_TEXT"

（xxx表示你的工程名称、-arch后所带参数为发生crash的机器的框架类型，iOS设备机型对应框架类型这里不做介绍）。

输入指令后所得结果如下：

Load command 1

cmd LC\_SEGMENT\_64

cmdsize 952

segname \_\_TEXT

vmaddr 0x0000000100000000

vmsize 0x00000000000b8000

fileoff 0

filesize 753664

maxprot 0x00000005

initprot 0x00000005

nsects 11

flags 0x0

其中 vmaddr 0x0000000100000000 地址便为我们所要的app运行起始地址。

### 二. 计算symbol address

symbol address = 地址偏移量＋起始地址。

起始地址上面我们已经得到，地址偏移量就是单行信息中“+”号后面的数值。

例如：

3 BaiduStatApp 0x000000010009e824 BaiduStatApp + 288804

上面的这行信息中288804就是地址偏移量。将288804转化为十六进制为：0x46824

计算：symbol address = 0x46824＋0x0000000100000000 = 0x0000000100046824

### 三. 解析symbol address

输入指令：

xcrun atos --arch arm64 -o xxx.app/xxx 0x0000000100046824

得到如下解析结果：

-[BaiduViewController viewDidLoad:] (in BaiduStatApp) (BaiduViewController.m:272)

结果中包含所在类名、函数名、文件名及对应行数。

## 第二节 使用load address相对偏移解析（方法二）

### 一． 计算load address相对偏移

例如：

3 BaiduStatApp 0x000000010009e824 BaiduStatApp + 288804

将288804转化为十六进制为：0x46824

计算：load address = 0x000000010009e824 - 0x46824 ＝ 0x0000000100058000

### 二． 使用load address解析

输入指令：

xcrun atos --arch arm64 -o xxx.app/xxx -l 0x0000000100058000 0x000000010009e824

得到如下解析结果：

-[BaiduViewController viewDidLoad:] (in BaiduStatApp) (BaiduViewController.m:272)

结果中包含所在类名、函数名、文件名及对应行数。与方法一结果相同。

# 联系我们

感谢您的阅读,如果有问题请 email 我们。 邮箱: [apptongji@baidu.com](mailto:apptongji@baidu.com)