

Android Native进程内存泄露检测

++ 主站

2016年12月17日 16:16:04

823



0



版权声明：本文为博主原创文章，未经博主允许不得转载。



Android Native进程内存泄露检测



简介



对于**Android**的native进程，Android源码中的Bionic库提供了一个很棒的API，get_malloc_leak_info用来检测Native代码内存泄露。

相关原理

相关的Property

使用这个API需要设置libc.debug.malloc这个property。这个property用来控制malloc信息的debug等级。在使用这个API之前，把libc.debug.malloc设置为1。关于这个property的说明可以在bionic/libc/bionic/malloc_debug_common.cpp中找到。下面代码给出其中的片段。

```
1 //when libc.debug.malloc enviroment variable value than  
2 //zero
```



forestcell

+ 关注

原创

83

粉丝

143

喜欢

22

码云

未开通



便宜智能手机



他的最新文章

更多文章

Linux-Bash技巧——字符串和base64互转

Android 中的dm-verity原理分析

TrustZone——市场普及——TEE组成部分浅析

基于TEE的安全系统有哪些？

SGX——多年的技术2016年终于有了SDK
下载

2
3

API:get_malloc_leak_info

在设置完property后，当在程序中调用malloc分配内存的时候，系统会调用到leak_malloc()这个API，这个API定义



在bionic/libc/bionic/malloc_debug_leak.c。leak_malloc()与普通malloc()的区别在于leak_malloc()会在真正分配的内存

0

空间前面会分配一段额外的空间来存储分配的内存信息。简而言之，如果使用malloc(4)（leak_debug版，系统



会分配4个字节的内存，然后在这个4字节内存的前面分配了一个头。所以整个malloc(4)产生的内存就像这样：



AllocationEntry | space[4bytes]].



这个AllocationEntry记录了malloc的call stack，内存大小还有一些其他的信息。它是存储在一个hashtable里的。可

以在malloc_debug_common.h这个文件中找到这个Hashtable的定义。以下选取了部分代码

```
1 struct HashEntry{
2     ....
3     size_t size;
4     size_t allocations;
5     ....
6 }
7
8
9 struct HashTable{
10     pthread_mutex_t lock;
11     size_t count;
12     HashEntry* slots[HASHTABLE_SIZE];
13 }
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

CPP	14篇
CSharp	7篇
GNU	5篇
项目管理	1篇
Android	8篇

展开

文章存档

2017年1月	1篇
2016年12月	3篇
2016年11月	1篇
2016年5月	2篇
2016年1月	1篇

展开

他的热门文章

TrustZone——Android的救星？

22210

Android文件系统保护——dmverity

12087

ARM——开发工具—编译器

10420

TrustZone——基础开发—Hello World

9058

ARM——操作系统—最小操作系统

7979

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册



11
12
13

这篇文章，不讨论如何去记录这些malloc信息，只概述一下如何去使用这项技术。这里有一篇文章很清楚的讲解了记录这些malloc信息的原理。



Android中native进程内存泄露的调试技巧



知道 leak_malloc 如何工作之后，那么这个API



```
1 void get_malloc_leak_info(uint8_t** info, size_t* overallSize, size_t* infoSize, size_t* totalMemory, size_t* backtraceSize)
```



就很容易使用了。它被定义在bionic/libc/bionic/malloc_debug_common.cpp这个文件中。

```
1  /***info" is set to a buffer we allocate
2  /***overallSize" is set to the size of "info" buffer
3  /***infoSize" is set to the size of single entry
4  /***totalMemory" is set to the sum of all allocations we're tracking; does
5  //not include heap overhead
6  /***backtraceSize" is set to the maximum number of entries in the back trace
```

在实际使用中，我们需要关注的是info指针，这个指针记录了malloc信息，从中解析后我们可以得到这些malloc大小和调用堆栈等等。

下面代码演示了如何去解析info指针

快速开发平台



联系我们



请扫描二维码联系客服

✉ webmaster@csdn.net

☎ 400-660-0108

🗣 网站客服

关于 招聘 广告服务 阿里云

©2018 CSDN 京ICP证09002463号

经营性网站备案信息

网络110报警服务

中国互联网举报中心

北京互联网违法和不良信息举报中心



0



```
4  size_t infoSize = 0;
5  size_t totalMemory = 0;
6  size_t backtraceSize = 0;
7  get_malloc_leak_info(&info, &overallSize, &infoSize, &totalMemory, &backtraceSize);
8  if (info) {
9      AllocEntry * entries = new AllocEntry[count];
10     for (size_t i = 0; i < count; i++) {
11         e->size = *reinterpret_cast<size_t *>(ptr);
12         ptr += sizeof(size_t);
13         e->dups = *reinterpret_cast<size_t *>(ptr);
14         ptr += sizeof(size_t);
15         e->backtrace = reinterpret_cast<intptr_t *>(ptr);
16         ptr += sizeof(intptr_t) * backtraceSize;
17     }
18     ....
19     1
20     2
21     3
22     4
23     5
24     6
25     7
26     8
27     9
28     10
29     11
30     12
31     13
32     14
33     15
34     16
35     17
36     18
```

如何在实际工程中使用

以在CameraService中debug为例

代码/编译部分

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册



```
dumpMemoryAddresses(int fd)
```

```
1
2
```

这个API封装了get_malloc_leak_info，它可以直接用来检测内存泄露,并将一次dump解析的信息放在了一个文件里。



举个例子，开发者想要去检测camera在preview时候的内存泄露。那么把这个API放在CameraService 的connect()，

那么每次当进入preview的时候都会去call connect()这个函数，此时他都会记下当前一次的内存情况。通过调用多



次，dump出不同时间段该进程的内存使用情况，就可以判断出是否存在内存泄露。



The MemoryTrackUtil.cpp 这个文件也可以在源码目录下这个位置找到: /android/frameworks/av/media/libmedia。



以下代码片段演示了如何使用dumpMemoryAddresses (int fd)

```
1 status_t CameraService::connect(...){
2     ....
3     Int fd = open(...);
4     dumpMemoryAddresses(fd);
5     ....
6 }
1
2
3
4
5
6
```

这样malloc的信息就会被保存在fd里，修改了这个文件之后，重新编译libcameraservice.so然后替换掉system.img中的.so

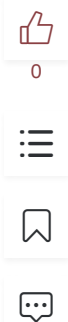
运行时使用方法

1.如果system.img中不存在libc malloc debug leak.so，那么把它push到/system/lib中并修改权限。

3.kill需要检测的进程，使这个property设置后生效。

这三部之后，camera在每次进入preview的时候都会去记录call stack和malloc的大小。

dump出来的文件如以下的格式:



```
size 262888, dup 1, 0x769a2032, 0x76ed9cce, 0x76ec991a, 0x76f52034, 0x76f62af8,
size 262144, dup 1, 0x769a2032, 0x76ed9cce, 0x76a824d4, 0x76a5e99a, 0x76bf39c2,
size 172036, dup 3, 0x769a2032, 0x76ed9cce, 0x764378c6, 0x7643791e, 0x76414912,
size 172036, dup 3, 0x769a2032, 0x76ed9cce, 0x764378c6, 0x7643791e, 0x76414912,
1
2
3
4
5
```

size代表malloc的内存大小。

dup值代表同一个地方分配的内存次数。对比dump出的不同时间段的文件，如果发现某一条malloc信息的dup越来越大，那么可以怀疑这个地方存在内存泄露。

最后一串字符串表示的是调用的堆栈。这个堆栈需要去减掉栈区间的起始位置，然后才能使用addr2line去查找这个调用堆栈。

举个例子：

```
size 262888, dup 1, 0x765a2032,
-----
765b1000-76688000 r-xp 00000000 b3:05 875 /system/lib/libc.so
1
2
3
4
```

那么实际主使用addr2line的stack值应该是 0x765a2032 - 765b1000

在MediaServer中调用这个API就很容易了

Android天然的已经在mediaserver中支持了这个功能。在MediaPlayerService.cpp中已经加入了dumpMemory 这个API。 并且以dumpsys的接口提供给用户使用，所以在对mediaserver相关进程进行debug的时候只需要如下几步即可：



0



```
setprop libc.debug.malloc 1
busybox killall -HUP mediaserver
dumpsys media.player -m
```

1

2

3

4



当然，如果需要dump不同时间段的信息，那么只需要写一个shell脚本就可以了。

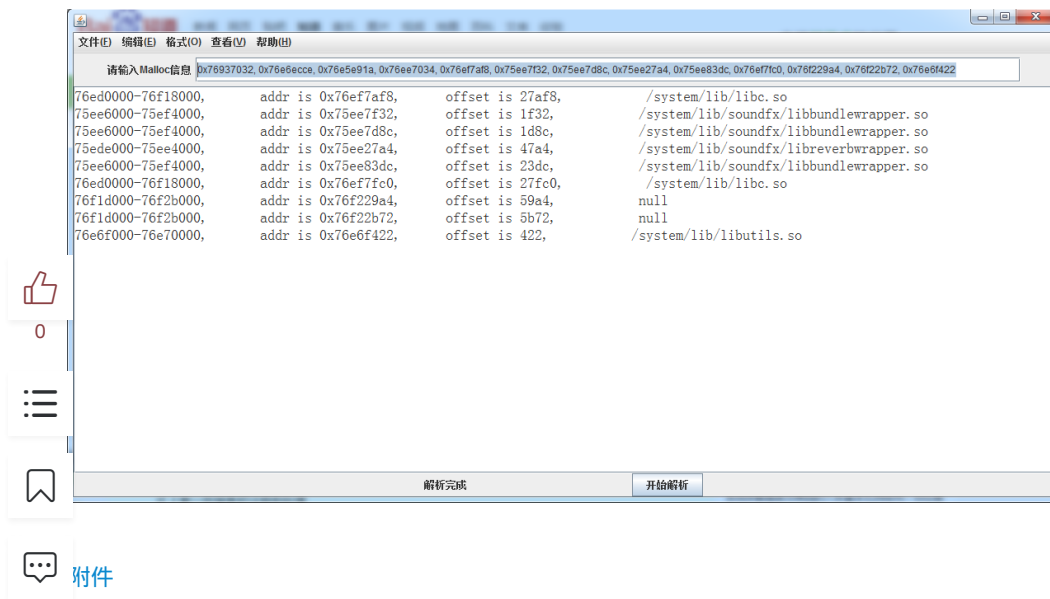
工具

手工计算堆栈信息是非常无聊的，所以我写了一个java程序(附件中)来去解析这个堆栈信息。

导入map文件。 将cat /proc/pid/maps所产生的字符串保存在一个文件里，这个文件就是map文件。

复制一条dump出来的malloc信息到文本框里

点击开始解析。



转自<http://blog.csdn.net/u011280717/article/details/51820268>

目前您尚未登录，请 [登录](#) 或 [注册](#) 后进行评论

android native 内存泄露检查 (libc.debug.malloc)

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

[登录](#)

[注册](#)



如何在安卓系统中侦测和调试内存泄露和越界



wlsfling

2017年03月28日 17:12

833

1.1 基本原理 使用bionc的libc初始时，会检测属性"libc.debug.malloc"，//android/bionic/libc/bionic/malloc_debug_common...



免费领取《海外游英语》，境外旅行说走就走

立即领取



调试malloc(堆越界)问题



sandform

2016年05月26日 16:24

1886

调试malloc(堆越界)问题 [DESCRIPTION] 有一类NE比较特殊，就是堆引起的异常(调用malloc申请的内存后使用不当引起)：1. 申请后多次释放 (doub...

android 系统内存检查



u011279649

2016年03月09日 07:35

811

1. Introduction Android对内存的使用包括内存泄漏和内存越界，内存泄漏会导致系统内存减少，最终分配不到内存，这样大的程序就不能运行，甚至系统没有内存而崩溃。Andro...

Android中native进程内存泄露的调试技巧（一）-- libc debug

libc.debug.malloc // 1 - For memory leak detections. // 5 - For filling allocated / freed memory...



agwtpcbox

2016年11月30日 14:50

1259

区块链概念股大揭秘！这些股值得入手！

【网易官方股票交流群】添加微信好友，进群免费领牛股→



(转)记一次内存优化的分享



wangbin_jxust

2015年05月18日 09:43

623

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册



Android开发——分析Native层内存泄漏



zjd934784273

2017年04月03日 13:28

📖 579

版权声明：本文为博主原创文章，未经博主允许不得转载。 目录(?)[+] Android开发——使用DDMS分析Native层内存泄漏 针对Java层的内存...

Android native 内存泄露检查 (libc.debug.malloc)

内存泄漏和内存越界



u010481276

2018年01月03日 11:27

📖 93

Android中native进程内存泄露的调试技巧



broadview2006

2013年01月31日 09:47

📖 3096

Android中native进程内存泄露的调试技巧 红狼博客 代码基于Android2.3.x版本 Android为Java程序提供了方便的内存泄露信息工具（如MAT），便于查找。但是，对于...

Android中native进程内存泄露的调试技巧（一）



L_nan

2015年02月04日 18:03

📖 8078

基于Android5.0版本 Android为Java程序提供了方便的内存泄露信息和工具（如MAT），便于查找。但是，对于纯粹C/C++ 编写的native进程，却不容易查找内存泄露。传统的C/C...

英语文档看不懂？教你一个公式秒懂英语！

跨界老码农教你学英语，带你有效提升阅读英文技术文档的能力→



Android Native内存泄漏诊断



yellowcath

2017年09月25日 15:08

📖 685

Android Native内存泄漏诊断1、基础诊断方法特点：操作简单，但只能判断是否有泄漏，但需使用者自行判断泄漏在哪里命令行方式adb shell dumpsys meminfo vStudio....

Android开发——分析Native层内存泄漏



u010307119

2016年11月12日 23:03

📖 5491

Android开发——使用DDMS分析Native层内存泄漏针对Java层的内存泄漏，Android提供了方便的内存泄漏检测工具，例如M

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册



- Android Native进程内存泄露检测

u011280717

2016年07月04日 13:26

2882

Android Native进程内存泄露检测简介对于Android的native进程，Android源码中的Bionic库提供了一个很棒的API，get_malloc_leak_info用来检测Native...

Android中native进程内存泄露的调试技巧（一）

L_nan

2015年02月04日 18:03

8078

Android5.0版本 Android为Java程序提供了方便的内存泄露信息和工具（如MAT），便于查找。但是，对于纯粹C/C++ 编写的native进程，却不容易查找内存泄露。传统的C/C...

Android中native进程内存泄露的调试技巧

lyuan1314

2013年12月19日 16:35

1403

: http://www.redwolf-blog.com/?p=1233

教你一招搞定背单词难题！

巧记单词so easy!

你抢《英语单词速记》

Pest

单词 搞定它

限时 抢书

免费申请

Android开发——分析Native层内存泄漏

u010307119

2016年11月12日 23:03

5491

Android开发——使用DDMS分析Native层内存泄漏针对Java层的内存泄漏，Android提供了方便的内存泄漏检测工具，例如MAT、LeakCanary。但对于native层开发，要追查C/...

Android Native内存泄漏诊断

yellowcath

2017年09月25日 15:08

685

Android Native内存泄漏诊断1、基础诊断方法特点：操作简单，但只能判断是否有泄漏，但需使用者自行判断泄漏在哪里命令行方式adb shell dumpsys meminfo vStudio....

delphi内存泄露查找工具之MemProof教程

dongyue786

2012年10月23日 14:07

1591

MemProof教程简介 MemProof（内存清道夫）是AutomatedQA出品的一款非常不错的检测内存泄漏和资源泄漏的免费调试工具，适合于WIN32平台下使用DELPHI/C...

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录

注册

×

http://blog.csdn.net/forestcell/article/details/53708706

11/11