

谦虚的天下

博客园 首页 新随笔 联系 订阅 管理

随笔 - 88 文章 - 10 评论 - 1402

Android学习系列(29)--App调试的几个命令实践

在Android的应用开发中，我们会用到各种代码调试；其实在Android的开发之后，我们可能会碰到一些随机的问題，如cpu过高，内存泄露等，我们无法简单的进行代码调试，我们需要一个系统日志等等，下面我把握工作中碰到的几个常用命令和方法给大家演示实践一下。

1.logcat命令

这个命令最简单常用，可查看帮助，我不多说，如果需要打印时间，加参数-v time

```
1 | adb logcat -v time
```

2.bugreport命令

这个命令也非常简单，但是在实际应用中非常有用，会有从开机之后详细的dumpsys,dumpstate和logcat信息，是一份完整的日志记录。对分析用户行为，异常信息，系统状态有很大的参考作用。一般我们会把bugreport导出到电脑上分析。

```
1 | adb bugreport > xxx.log
```

我再次强调，bugreport里面包含丰富的系统和用户信息，它是其他很多命令输出的结果的记录，非常有用。

3.dumpsys命令

这个查看系统信息，用的还是比较多的。

```
1 | dumpsys [options]
2 |             meminfo 显示内存信息
3 |             cpuinfo 显示CPU信息
```

公告

昵称：谦虚的天下
园龄：9年5个月
荣誉：推荐博客
粉丝：2159
关注：28
+加关注

< 2012年5月 >						
日	一	二	三	四	五	六
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

搜索

```
4      account 显示accounts信息
5      activity 显示所有的activities的信息
6      window 显示键盘，窗口和它们的关系
7      wifi 显示wifi信息
```

例如查看某个程序内存信息:

```
1  #查看应用com.tianxia.test的内存使用情况
2  adb shell dumpsys meminfo com.tianxia.test
```

效果图如下：

常用链接

[我的随笔](#)

[我的评论](#)

[我的参与](#)

[最新评论](#)

[我的标签](#)

我的标签

[Android \(73\)](#)

[Android学习系列 \(42\)](#)

[FastObject \(20\)](#) [ExtJs2.0 \(16\)](#)

[Android设计模式系列 \(13\)](#)

[Android拓展系列 \(11\)](#)

[设计模式 \(10\)](#) [Ext \(4\)](#)

[Extjs \(4\)](#) [个人框架 \(3\)](#) [更多](#)

随笔分类⁽⁷⁰⁾

[Android框架源码剖析系列](#)

[Android设计模式系列\(13\)](#)

[Android拓展系列\(12\)](#)

[Android陷阱系列](#)

[Android学习系列\(39\)](#)

[Android应用\(6\)](#)

随笔档案⁽⁸⁸⁾

[2016年3月 \(1\)](#)

[2015年11月 \(1\)](#)

[2015年9月 \(3\)](#)

[2015年3月 \(1\)](#)

[2015年1月 \(1\)](#)

[2014年12月 \(1\)](#)

```
D:\>adb shell dumpsys meminfo com.tianxia.test
Applications Memory Usage (kB):
Uptime: 1454739 Realtime: 1454739

** MEMINFO in pid 644 [com.tianxia.test] **
      Pss      Shared   Private  Heap    Heap    Heap
      ---      ---      ---      Size    Alloc    Free
Native       942       1180       856     3632     3187     240
Dalvik      1148      11564       556     9347     9218     129
Cursor         0         0         0
Ashmem         0         0         0
Other dev        4         0         0
.so mmap       623      1848       200
.jar mmap        0         0         0
.apk mmap       75         0         0
.ttf mmap        0         0         0
.dex mmap        4         0         0
Other mmap     345        20        12
Unknown       529       680       488
TOTAL       3670     15292     2112    12979    12405     369

Objects
  Views:          20      ViewRootImpl:      0
  AppContexts:     3      Activities:        1
  Assets:          2      AssetManagers:     2
  Local Binders:   1      Proxy Binders:     7
  Death Recipients: 0
  OpenSSL Sockets: 0

SQL
  heap:           0      MEMORY_USED:      0
  PAGECACHE_OVERFLOW: 0      MALLOC_SIZE:      0

Asset Allocations
  zip:/data/app/com.tianxia.test-1.apk:/resources.arsc: 1K
```

里面的信息很有价值，尤其对于分析内存泄露，内存溢出都有极大的作用。

4.top命令

这个查看cpu信息太方便了。

```
1 | top -m 5 -t
```

我们看看效果图，其中按cpu大小列出5个进程列表。

- 2014年9月 (2)
- 2014年7月 (1)
- 2014年5月 (2)
- 2014年4月 (4)
- 2013年4月 (2)
- 2012年12月 (5)
- 2012年11月 (1)
- 2012年10月 (1)
- 2012年9月 (1)
- 2012年8月 (1)
- 2012年7月 (2)
- 2012年6月 (1)
- 2012年5月 (2)
- 2012年4月 (2)
- 2012年3月 (1)
- 2012年2月 (2)
- 2012年1月 (1)
- 2011年12月 (1)
- 2011年9月 (4)
- 2011年8月 (9)
- 2011年7月 (4)
- 2011年6月 (6)
- 2011年5月 (4)
- 2011年4月 (5)
- 2008年8月 (10)
- 2008年7月 (2)
- 2008年6月 (4)

积分与排名

积分 - 389832
排名 - 351

最新评论

1. Re:Android学习系列(33)--...

```

D:\>adb shell
# top -m 5 -t
top -m 5 -t

User 37%, System 62%, IOW 0%, IRQ 0%
User 149 + Nice 0 + Sys 244 + Idle 0 + IOW 0 + IRQ 0 + SIRQ 0 = 393

  PID   TID PR CPU% S      VSS      RSS PCY UID      Thread      Proc
  644   644  0  85% R 156732K  31660K fg app_41  om.tianxia.test com.tianxia.t
est
  668   668  0  11% R  1048K   504K fg root    top          top
  36    64  0   0% S  27272K 10300K fg system  SurfaceFlinger /system/bin/s
urfaceflinger
  141   141  0   0% S 165824K 46260K fg system  ndroid.systemui com.android.s
ystemui
  89    121  0   0% S 236172K 58696K fg system  er$SensorThread system_server

```

com.tianxia.test的cpu过高，会导致手机发烫。同时利用这个信息，可以监控应用cpu的使用，以调整优化代码。

5.配置文件local.prop

目前网上没有查到local.prop的配置使用，工作中本人只使用过如下：

```
1 | log.tag.SQLiteStatements=VERBOSE log.tag.SQLiteTime=VERBOSE
```

把上述文本加到/data/local.prop中，如果没有这个文件自行创建。然后重启手机，就能看到每个应用详细的查询数据库的sql语句信息，对于调试数据库，分析和优化数据库sql异常非常有用。

6.分析手机发烫

下面我们来实践一个例子，手机发烫太厉害，怎么找出问题？

首先我们写一个程序com.tianxia.test，死循环，核心代码如下：

```

1 | @Override
2 | public void onCreate(Bundle savedInstanceState) {
3 |     super.onCreate(savedInstanceState);
4 |     setContentView(R.layout.main);
5 |     while(true) {
6 |         System.currentTimeMillis();
7 |     }
8 | }

```

苹果安卓app，无需审核直接发布稳定有效，无需帐号一键安装，渠道配合快速修改上架市场，微信里直接安装安卓苹果APP快速上架各大APP应用商店！积分墙机刷,App冲刷榜，刷下载量评论,ASO应用商店优化.....

--网络技术开发者

2. Re:Android打包的那些事

问下，如果我当前项目有源码库工程依赖，那怎么对该源码库做混淆？

--petercao

3. Re:[Android应用]《幽默笑...

楼主你好，我发现 精选的hostname失效了啊，
www.kaiyuanxiangmu.com但是我看看你的博客都没问题
=qrcode&frwbqr=53 大神能否提供一下 新的api接口地址？感谢...

--aspirant

4. Re:记一次小团队Git实践(上)

@Ruo_Nan加了...

--菜鸟就的先飞

5. Re:Android学习系列(17)--...

請問該如何讓初始狀態就有選中的效果？

--nex314

阅读排行榜

1. Android学习系列(1)--为Ap...
2. Android学习系列(28)--App...
3. Android学习系列(2)--App自..
4. Android学习系列(33)--App...
5. Android学习系列(9)--App列..

评论排行榜

这个应用打开后会一直获取系统时间，因为在主线程，肯定导致应用ANR，也会一直浪费系统cpu，手机发热，我们运行它。

假设我们不知道上述代码，我们来找到这个问题：

(1).找到发烫的应用。

使用top命令：

```
1 | top -m 5 -t
```

```
D:\>adb shell
# top -m 5 -t
top -m 5 -t

User 37%, System 62%, IOW 0%, IRQ 0%
User 149 + Nice 0 + Sys 244 + Idle 0 + IOW 0 + IRQ 0 + SIRQ 0 = 393

  PID    TID  PR CPU% S   VSS      RSS PCY UID      Thread                Proc
  644    644   0  85% R 156732K  31660K fg app_41  om.tianxia.test      com.tianxia.t
est
  668    668   0  11% R   1048K   504K fg root   top                  top
  36     64   0   0% S  27272K  10300K fg system SurfaceFlinger      /system/bin/s
urfaceflinger
  141    141   0   0% S 165824K  46260K fg system  ndroid.systemui      com.android.s
ystemui
  89    121   0   0% S 236172K  58696K fg system  er$SensorThread     system_server
```

一看是com.tianxia.test占用85%的cpu，原来是这家伙在捣鬼。进程ID是644，这个后面我们有用。

(2).分析发烫的应用进程在干嘛

需要用到linux下的strace命令，但是android是没有集成这个命令的，android版本的下载地址：

<http://benno.id.au/android/strace>

下载完成后，上传到手机中：

我们adb push strace /system/bin，在模拟器上是上传到/system/bin会报out of memory错误，我们也可以上传到/data目录下，如果没有执行权限，还需要chmod 777 strace.

strace命令有很多参数，直接执行strace会显示使用说明:

1. Android学习系列(2)--App自..
2. ExtJs2.0学习系列(1)--Ext....
3. ExtJs2.0学习系列(4)--Ext.F..
4. ExtJs2.0学习系列(2)--Ext....
5. ExtJs2.0学习系列(8)--Ext.F..

推荐排行榜

1. Android学习系列(1)--为Ap...
2. Android学习系列(2)--App自..
3. Android学习系列(27)--App...
4. Android学习系列(7)--App轮..
5. Android学习系列(6)--App模..

```

D:\>adb shell
# cd data
cd data
# strace
strace
usage: strace [-dffhiqrtttTvVxx] [-a column] [-e expr] ... [-o file]
              [-p pid] ... [-s strsize] [-u username] [-E var=val] ...
              [command [arg ...]]
or: strace -c [-e expr] ... [-O overhead] [-S sortby] [-E var=val] ...
              [command [arg ...]]
-c -- count time, calls, and errors for each syscall and report summary
-f -- follow forks, -ff -- with output into separate files
-F -- attempt to follow vforks, -h -- print help message
-i -- print instruction pointer at time of syscall
-q -- suppress messages about attaching, detaching, etc.
-r -- print relative timestamp, -t -- absolute timestamp, -tt -- with usecs
-T -- print time spent in each syscall, -V -- print version
-v -- verbose mode: print unabbreviated argv, stat, termio[s], etc. args
-x -- print non-ascii strings in hex, -xx -- print all strings in hex
-a column -- alignment COLUMN for printing syscall results (default 40)
-e expr -- a qualifying expression: option=[!]all or option=[!]val1[,val2]...
      options: trace, abbrev, verbose, raw, signal, read, or write
-o file -- send trace output to FILE instead of stderr
-O overhead -- set overhead for tracing syscalls to OVERHEAD usecs
-p pid -- trace process with process id PID, may be repeated
-s strsize -- limit length of print strings to STRSIZE chars (default 32)
-S sortby -- sort syscall counts by: time, calls, name, nothing (default time)
-u username -- run command as username handling setuid and/or setgid
-E var=val -- put var=val in the environment for command
-E var -- remove var from the environment for command

```

其中-p参数输入的就是进程号，第一步中我们找到com.tianxia.test的进程ID是644，我们看看这个应用占用这么高的cpu在干嘛？

```
1 | strace -p 644
```

输出如下：


```
D:\>adb shell
# strace -p 644
strace -p 644
Process 644 attached - interrupt to quit
gettimeofday({1336924194, 295497}, NULL) = 0
gettimeofday({1336924194, 296201}, NULL) = 0
gettimeofday({1336924194, 297581}, NULL) = 0
gettimeofday({1336924194, 303550}, NULL) = 0
gettimeofday({1336924194, 305251}, NULL) = 0
gettimeofday({1336924194, 305969}, NULL) = 0
gettimeofday({1336924194, 306588}, NULL) = 0
gettimeofday({1336924194, 307262}, NULL) = 0
gettimeofday({1336924194, 307916}, NULL) = 0
gettimeofday({1336924194, 308586}, NULL) = 0
gettimeofday({1336924194, 309202}, NULL) = 0
gettimeofday({1336924194, 314080}, NULL) = 0
gettimeofday({1336924194, 317064}, NULL) = 0
gettimeofday({1336924194, 322571}, NULL) = 0
gettimeofday({1336924194, 323327}, NULL) = 0
```

它的系统调用一直是gettimeofday,一直输出这个,显然哪里一定进入死循环了,而且是获取时间的死循环,然后结合logcat和代码,定位这段代码(就是前面我们给出的那段代码了)解决这个bug。

7.采集手机的cpu运行情况.

有时使用日志我们很难针对性的获取我们想要的信息,我们可能需要写一些最简单的脚步放在手机里面执行。

如监控cpu占用的记录cpu_log.sh:

```
1  # !/system/bin/sh
2  #这个脚步比较粗糙,是这么个意思
3  file=/sdcard/cpu/cpu_info.log
4  rm $file
5  until [ 1 -gt 10000 ]
6  do
7  echo -e "\n\n\n\n\n-----" >> $file
8  date >> $file
9  top -m 5 -n 1 >> $file
10 sleep 3
11 done
```

每隔3s中就会把手机的cpu的信息写到sdcard的cpu目录下的cpu_info.log文件中,方便我们后续分析。

ps:使用方法是 push到data目录下,赋予可执行权限,在shell下执行即可。

8.采集某个应用的内存数据

这个实践和上面的脚本类似，只是命令不一样我另外单独列出来，因为这个有时候很有用。

比如，我们要采集com.tianxia.test的内存使用情况，分析它是不是会内存泄露，脚本类似：

```
1 #!/system/bin/sh
2 #这个脚本比较粗糙，是这么个意思
3 file=/sdcard/cpu/mem_info.log
4 rm $file
5 until [ 1 -gt 10000 ]
6 do
7 echo -e "\n\n\n\n\n-----">>$file
8 date >> $file
9 dumpsys meminfo com.tianxia.test >> $file
10 sleep 3
11 done
```

使用方法也是一样。

9.小结

零零碎碎的一直没有时间整理，有常用的也有不常用的，算是一些小技巧，感觉网上这方面的分享比较少，有时和朋友们谈起这些调试方法，特别是龙哥，硬是要求我今天写出来与大家分享，只好献丑，说不定对于解决一些疑难杂症有奇效，呵呵，想到什么写什么吧，想到几个写几个，也是一个学习的记录。

分类: [Android学习系列](#)

标签: [Android](#), [Android学习系列](#)

好文要顶

关注我

收藏该文



谦虚的天下

关注 - 28

粉丝 - 2159

荣誉：推荐博客

[+加关注](#)

9

0

« 上一篇 : [\[Android应用\]《花界》V1.0 正式版隆重发布！](#)

» 下一篇 : [Android拓展系列\(6\)--CM9源码下载和编译](#)

posted @ 2012-05-14 00:17 谦虚的天下 阅读(24706) 评论(15) 编辑 收藏

评论列表

#1楼

2012-05-14 15:01 Mr_Hannibal

这个貌似现在还没有用上哦，但是学习了！

支持(0) 反对(0)

#2楼

[楼主] 2012-05-14 15:07 谦虚的天下

@ Mr_Hannibal

呵呵，你会用到的

支持(0) 反对(0)

#3楼

2012-05-16 13:06 大浪淘沙001

楼主功力深厚，受教了！

支持(0) 反对(0)

#4楼

2012-05-22 09:21 天天_byconan

确实很有用，学习了！

支持(0) 反对(0)

#5楼

2012-05-25 09:00 爱你就要你

太喜欢这个博客了，希望能得到大大的指导

支持(0) 反对(0)

#6楼

2012-06-06 10:51 WillYan

adb logcat -s dalvikvm-heap

支持(0) 反对(0)

#7楼

2012-06-16 20:34 cmlanche-明

太牛了！

支持(0) 反对(0)

#8楼

2012-08-10 19:29 __Shadow

顶一个！

支持(0) 反对(0)

#9楼

2012-11-10 13:18 天籁の圖

感谢分享

支持(0) 反对(0)

#10楼

2013-05-08 20:25 Please Call me 小强

这不就是linux命令不。很有用

支持(0) 反对(0)

#11楼

2013-08-06 10:40 jackrex

楼主屌炸天啊

支持(0) 反对(0)

#12楼

2014-07-13 21:18 xdj117

楼主你好，问一个问题，就是strace命令我在pc端的调试环境下可以运行，但是我想再android APP编程的时候用exec调用这个命令，怎么试都不行，经检查，系统已root，方法没问题，其他命令例如top、ls可以运行。求楼主帮忙解答！小弟拜谢了！

支持(0) 反对(0)

#13楼 [楼主] 2014-07-16 17:21 谦虚的天下

@ xdj117

手机默认是没有strace命令的，是否push strace到手机了？

支持(0) 反对(0)

#14楼 2015-01-21 15:50 三道

已关注，谢谢楼主，楼主厉害！！

支持(0) 反对(0)

#15楼 2015-04-15 14:31 scue

还可以有 vmstat, df, netcfg

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【推荐】腾讯云上实验室 1小时搭建人工智能应用

【推荐】可嵌入您系统的“在线Excel”！SpreadJS 纯前端表格控件

【推荐】阿里云“全民云计算”优惠升级



最新IT新闻:

- 东芝向财团出售芯片交易推迟 因苹果不同意部分关键条款
 - 想要实现自动驾驶？高精度地图不可或缺
 - 智联招聘将退市美股，股东已批准合并协议
 - 腾讯官方回应微信卡死Bug：正在紧急修复
 - 微软推出三款新机器学习工具 帮助开发者打造AI应用
- » 更多新闻...



最新知识库文章:

- 如何阅读计算机科学类的书
 - Google 及其云智慧
 - 做到这一点，你也可以成为优秀的程序员
 - 写给立志做码农的大学生
 - 架构腐化之谜
- » 更多知识库文章...

