

昵称: 小学徒V
园龄: 2年1个月
粉丝: 185
关注: 9
[+加关注](#)

< 2015年5月 >

日	一	二	三	四	五	六
26	27	28	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

搜索

找找看

谷歌搜索

最新随笔

1. Appium的安装 - MAC平台

2. alias指令: 设置命令别名

3. Android学习——第一个NDK程序

4. Android学习——windows下搭建Cygwin环境

5. Android学习——windows下搭建NDK_r9环境

6. Win7&Ubuntu12.04 双系统引导问题

7. Android学习第三天-签名常用命令

8. Android学习第三天-打包常用命令

9. Android学习第二天-android常用命令

10. Android学习第一天-ADB常用命令

随笔分类(58)

Android(10)

Java EE(2)

Java SE(14)

Linux/Unix(9)

mac(1)

PHP(2)

Python(3)

WEB前端(4)

编程软件(2)

测试相关(1)

求职之道(4)

设计模式(2)

数据结构(1)

数据库(2)

算法之道(1)

随笔档案(40)

2014年11月 (1)

2014年10月 (1)

2013年12月 (3)

2013年11月 (5)

2013年10月 (2)

2013年9月 (3)

2013年8月 (5)

2013年7月 (3)

2013年6月 (5)

2013年5月 (4)

2013年4月 (3)

2013年3月 (5)

友情链接

ggjucheng

阿里云@java

迪迪@IOS

土豪@java

伍迷

小学徒51CTO博客

资源共享

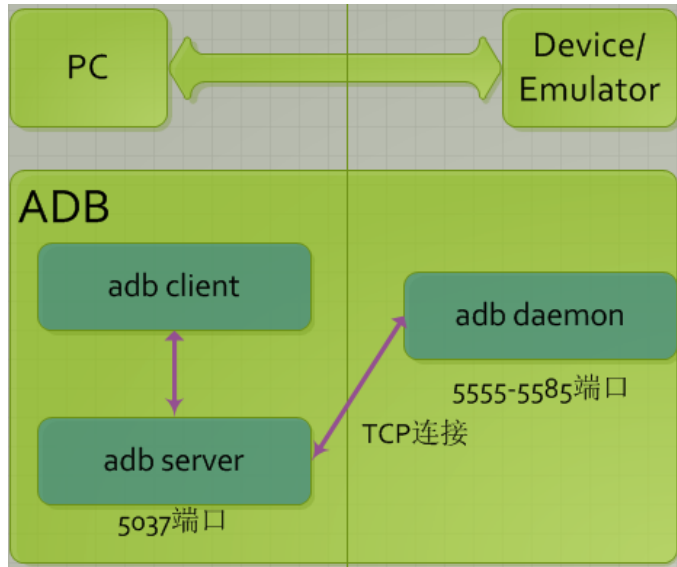
Android学习第一天-ADB常用命令

平时开发android应用 的时候，我们都会用到包含在Android SDK中一系列的工具，或许我们通过Eclipse去调用，又或许，我们自己通过打开终端进行手动输入并且执行，下面我们来一起学习下这些工具的开发使用吧。

1.Android Debug Bridge

1.1简介

Android Debug Bridge，我们一般简称为adb，主要存放在sdk安装目录下的platform-tools文件夹中，它是一个非常强大的命令行工具，通过这个工具你能够与你的android设备进行交互。同时这是一个包含了以下三部分的cs模式的程序：



1) adb client

从图中，我们知道client是运行在PC端的，每当我们发起一个adb命令的时候，就会开启一个client程序。当然，当我们开启DDMS或者ADT的时候，也会自动创建client。

当我们开启一个client的时候，它首先会去检测后台是否已经有一个server程序在运行着，否则会开启一个adb-server进程。

所有的client都是通过5037端口与adb-server进行通信的。

2) adb daemon (adb)

从图中，我们知道daemon是作为一个后台进程运行在模拟器/真实Android设备中的。

daemon使用端口的范围是5554-5585，每个模拟器/设备连接到PC端时，总会开启这么一个后台进程，并且为其分配了两个连续的端口，比如：

```
Emulator 1, console: 5554
Emulator 1, adb: 5555
```

也正因为每个设备都分一组两个端口，也已adb连接手机的最大数量为16。

说回端口的作用，在这两个端口中，其中偶数端口是用于server 与设备进行交互的，可以让server直接从设备中读取数据，而奇数端口是用来与设备的adb进行连接通信的。

2) adb server

从图中，我们同样可以知道，server也是作为一个后台的程序运行在PC端的，他负责管理client进程以及adb daemon之间的通信。

当一个server开启的时候，他会自动绑定并且监听5037端口，接收client通过该端口发送过来的命令。同时server还会对5555-5585间的奇数端口进行扫描，进行对已连接设备的定位。

1.2 常用的adb命令

首先我们先来介绍一下，adb命令的使用格式吧：

```
adb [-d|-e|-s <serialNumber>] <command>
```

adb是必须的，接下来方括号[]里面的内容不是必须的，最后才是我们需要执行的命令操作，例如 adb -s emulator-5554 install UCBrowser.apk (这是安装UC浏览器的意思)

积分与排名

积分 - 45757
排名 - 4154

最新评论

1. Re:Java的内存回收机制
3) 尽早释放无用对象;纠结了, 我以前网上搜【java手动释放】很多回答都是自动回收机制是java的优势不必要手动释放。将对象=null 之后也不回马上回收, 也要等到GC一段时间后才会计算是否该回收。.....

--koik

2. Re:Java的内存回收机制
我记得finalize这个方法是不可保证的

--wuhn

3. Re:Android学习——windows下搭建NDK_r9环境

非常nice的文章 有一点提出 就是会报很多not resolved type的错误 但是不影响程序运行 估计是eclipse自己的问题 治标不治本的解决方法: Project-> propertie S-.....

--jasonniu

4. Re:Android学习第一天-adb常用命令
adb shutdown 无效....

--coco+

5. Re:CVTE实习求职经历
这个公司, 工作强度太大, 这个强度可以媲美游戏公司, 但是游戏公司与他相比, 绝对是有优势的。。。尤其是工作一二年后再出来。。

--Ryan_1715575332

阅读排行榜

- 1. Android学习第一天-adb常用命令(8367)
- 2. Java的内存回收机制(6705)
- 3. 腾讯2013年实习生笔试题目 (附答案) (6580)
- 4. CVTE实习求职经历(6030)
- 5. 腾讯实习初面经过(3167)
- 6. 小学徒成长系列—线程同步、死锁、线程池(2192)
- 7. Android学习——windows下搭建Cygwin环境(2188)
- 8. Android学习第二天-android常用命令(1779)
- 9. 使用MYSQL命令直接导入导出SQL文件(1743)
- 10. Java数组及其内存分配(1704)

评论排行榜

- 1. Java的内存回收机制(22)
- 2. 腾讯2013年实习生笔试题目 (附答案) (21)
- 3. 腾讯实习初面经过(17)
- 4. CVTE实习求职经历(16)
- 5. Java对象创建方式及JVM对字符串处理(7)
- 6. 深入了解JVM—内存区域(6)
- 7. Java基础类型总结(6)
- 8. Ubuntu环境搭建系列—WPS/LAMP/Python篇(6)
- 9. Win7&Ubuntu12.04 双系统引导问题(6)
- 10. 小学徒进阶系列—JVM对String的处理(6)

推荐排行榜

- 1. Java的内存回收机制(33)
- 2. 腾讯2013年实习生笔试题目 (附答案) (9)
- 3. Android学习第一天-adb常用命令(7)
- 4. CVTE实习求职经历(7)
- 5. 小学徒成长系列—String关键源码解析(7)

Android学习第一天-adb常用命令 - 小学徒V - 博客园

下面先介绍下, 前面几个参数的含义吧:

- d: 让唯一连接到该PC端的真实安卓设备执行命令, 如果发现USB中连接有多部设备, 将会报错
- e: 让唯一连接到该PC端的模拟器执行命令, 如果发现开启了多个模拟器, 将会报错
- s: 通过设备的序列号进行指定设备执行命令

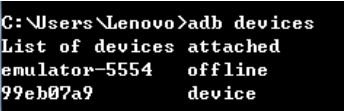
如果设备只连接有一个设备或者一个模拟器的时候, 可以不用声明这三个参数, adb默认会让这部唯一连接到的设备进行命令执行。

ok, 下面开始讲解常用的命令吧:

1) 查看当前PC端连接有多少设备:

```
adb devices
```

终端中的执行结果为:



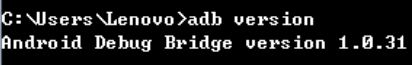
从上图, 我们看到了设备有的两种状态, 其实执行该命令, 可能返回的状态有三种:

- I. device 设备已经成功连接到了adb-server
- II. offline 设备并没有连接到adb或者没有响应
- III. no device 并没有设备/模拟器连接

2) 查看adb的版本

```
adb version
```

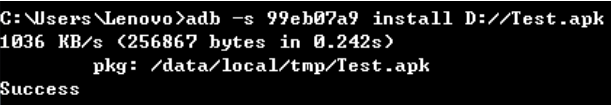
终端中的执行结果为:



3) 给设备进行软件的安装

```
adb -s <serialNumber> install <path-to-apk>
eg:
adb -s 99eb07a9 install D://Test.apk
```

终端中的执行结果为:



除了上面这种方法, 我们还有另外一个命令:

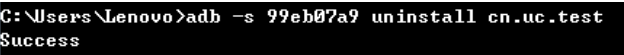
```
adb -s <serialNumber> shell pm install [options] <PATH>
```

其中[options]中最常用的选项是-r, 代表着本次安装是重新安装, 会保留用户数据。

4) 卸载设备中已经安装的软件

```
adb -s <serialNumber> uninstall <pkg_name>
eg:
adb -s 99eb07a9 uninstall cn.uc.test
```

终端中的执行结果为:



除了上面这种方法, 我们还有另外一个命令:

```
adb -s <serialNumber> uninstall [options] <PACKAGE>
```

这里的options是-k, 加上该参数表明卸载软件后依旧保留用户数据

5) 将数据从设备复制到PC中

```
adb -s <serialNumber> pull <remote> <local>
eg:
adb -s 99eb07a9 pull /sdcard/stericson-ls D://
```

终端的执行结果:

6. 腾讯实习初面经过(6)
7. Java基础类型总结(6)
8. 小学徒成长系列—线程同步、死锁、线程池(5)
9. 对象与内存管理(4)
10. Ubuntu环境搭建系列—WPS/LAMP/Python篇(4)

```
C:\Users\Lenovo>adb -s 99eb07a9 pull /sdcard/stericson-ls D://
2774 KB/s (849476 bytes in 0.299s)
```

在D://盘中我们也可以发现stericson-ls文件

6) 讲数据从PC端复制到设备中

```
adb -s <serialNumber>push <local> <remote>
eg:
adb -s 99eb07a9 push d://stericson-ls /sdcard/
```

终端的执行结果:

```
C:\Users\Lenovo>adb -s 99eb07a9 push d://stericson-ls /sdcard/
2132 KB/s (849476 bytes in 0.389s)
```

7) 获取连接到手机的序列号

```
adb get-serialno
```

终端的执行结果:

```
C:\Users\Lenovo>adb get-serialno
99eb07a9
```

8) 获取设备的连接状态

```
adb -s <serialNumber> get-state
eg:
adb -s 99eb07a9 get-stat
```

终端的执行结果:

```
C:\Users\Lenovo>adb -s 99eb07a9 get-state
device
```

9) 等待设备连接成功

```
adb wait-for-device
```

该命令将会阻塞一直到默认连接的那部机器状态变为device,也就是连接成功。

10) adb-server 的开启与关闭

```
adb kill-server
adb start-server
```

终端的执行结果:

```
C:\Users\Lenovo>adb kill-server

C:\Users\Lenovo>adb start-server
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
```

有时候我们开启server会失败,这是因为我们装了的手机助手,豌豆荚或者其他软件自带有自己的adb,导致端口冲突开启失败。主要关掉那些软件并且杀掉那些软件自带的adb进程重来即可。

1.2 手机执行Shell命令

众所周知,Android的内核本身就是Linux,所以Android本身也支持Linux命令,但是因为是手机系统,所以进行了一定的删减,部分命令不支持,而不支持的部分,我们可以通过安装busybox这个软件进行填补。

下面我们讲解下如何在手机中执行Shell命令:

1) 每次都加上adb shell来进行命令的发送与执行

```
adb shell <command>
```

2) 先通过adb shell进入手机Shell执行状态,然后直接输入Linux命令,如图:

```
C:\Users\Lenovo>adb shell
shell@android:/ $ ls /data/local/tmp
ls /data/local/tmp
init.rc
screencap_4.1
tcpdump
shell@android:/ $
```

1.3 Activity Manager 常用命令与参数说明

一般情况下,我们最常用它来进行app的开启,并且传递一定的参数过去,比如:

```
adb shell am start -a android.intent.action.VIEW -n cn.uc.test/.MainActivity -d
http://www.baidu.com
```

详情的可以参考: <http://developer.android.com/tools/help/adb.html#IntentSpec>

在这里就不做太过详细和重复的说明了

1.4 Package Manager 常用命令与参数说明

1) 使一个apk处于禁用状态, 相当于卸载了:

```
disable <PACKAGE_OR_COMPONENT>
其中 PACKAGE_OR_COMPONENT的格式是 package/class, 如 com.UCMobile/.main.UCMobile
eg:
adb shell su disable com.UCMobile/.main.UCMobile
```

终端执行结果:

```
C:\Users\Lenovo>adb shell su pm disable com.UCMobile/.main.UCMobile
Component {com.UCMobile/com.UCMobile.main.UCMobile} new state: disabled
```

同时我们在手机上发现, UC浏览器的图标不见了

2) 让一个apk从禁用状态恢复过来

```
enable <PACKAGE_OR_COMPONENT>
其中 PACKAGE_OR_COMPONENT的格式是 package/class, 如 com.UCMobile/.main.UCMobile
eg:
adb shell su enable com.UCMobile/.main.UCMobile
```

终端执行结果:

```
C:\Users\Lenovo>adb shell su pm enable com.UCMobile/.main.UCMobile
Component {com.UCMobile/com.UCMobile.main.UCMobile} new state: enabled
```

同时我们从手机上发现, UC浏览器的图标又回来啦。

3) 查看一个apk所在的路径

```
pm path <Package>
eg:
adb shell pm path com.UCMobile
```

终端执行结果:

```
C:\Users\Lenovo>adb shell pm path com.UCMobile
package:/data/app/com.UCMobile-1.apk
```

4) 查看手机中装的所有的包

```
pm list packages
```

具体更多的详情, 可以参考: <http://developer.android.com/tools/help/adb.html#IntentSpec>

再次不做过多的说明

1.5其他常用的adb命令

1) 屏幕录像

例如, 屏幕开始录像并且储存在/sdcard中, 同时名字为demo.mp4

```
adb shell screenrecord /sdcard/demo.mp4
```

2) 关闭设备请求, 开启设备

```
adb shell stop
adb shell start
```

3) 设备关机以及重启

```
adb reboot
adb shutdown
```

转载请注明出处: <http://www.cnblogs.com/xiaoxuetu/>, 谢谢合作



哈喽, 大家好! 我是小学徒V。 您的支持是我无限的动力, 在此非常感谢您阅读完本篇文章。