



【风暴兄弟连】 ADB 命令详解

【目录】

【安装调试】	2
【常用命令详解】	3
1、查看设备情况	3
2、安装软件.....	4
3、卸载软件.....	5
4、进入设备或模拟器的 shell.....	5
5、发布端口.....	6
6、从电脑上发送文件到设备.....	6
7、从设备上下载文件到电脑.....	7
8、查看 bug 报告.....	7
9、在 adb shell 下可查看手机系统的具体命令	7
10、ubuntu 使用时出现的问题	8
11、使用 adb 进行 recovery 的过程.....	9
【ADB 命令列表】	9
Adb 命令详表.....	9
Adb 命令部分中文表.....	11

ADB 全称 Android Debug Bridge , 是 android sdk 里的一个工具 , 用这个工具可以直接操作管理 android 模拟器或者真实的 andriod 设备它的主要功能有 :

- 运行设备的 shell(命令行)
- 管理模拟器或设备的端口映射
- 计算机和设备之间上传/下载文件
- 将本地 apk 软件安装至模拟器或 android 设备

ADB 是一个 “客户端-服务器端” 程序 , 其中客户端就是用来操作

的电脑，服务器端是 android 设备，实体机器或者虚拟机。

【安装调试】

在此，需要用到的软件有：

- 1、 android_usb_windows，设备连接电脑的驱动包
- 2、 ADB 的工具包或者整个 SDK，adb 操作的工具软件

驱动的安装在此就不说了，驱动安全成功后的信息应该是这样的：



如果下载的是 ADB 工具包，需要做的只是将“adb.exe”和“AdbWinApi.dll”两个文件放到“C:\Windows\System32”里即可。想知道有没有成功，可以这样操作：

- 1、在搜索框输入 CMD，进入 DOS 界面



- 2、输入 adb，如果可以的话，可以出现一大堆的东西

```
CA. 管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7600]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

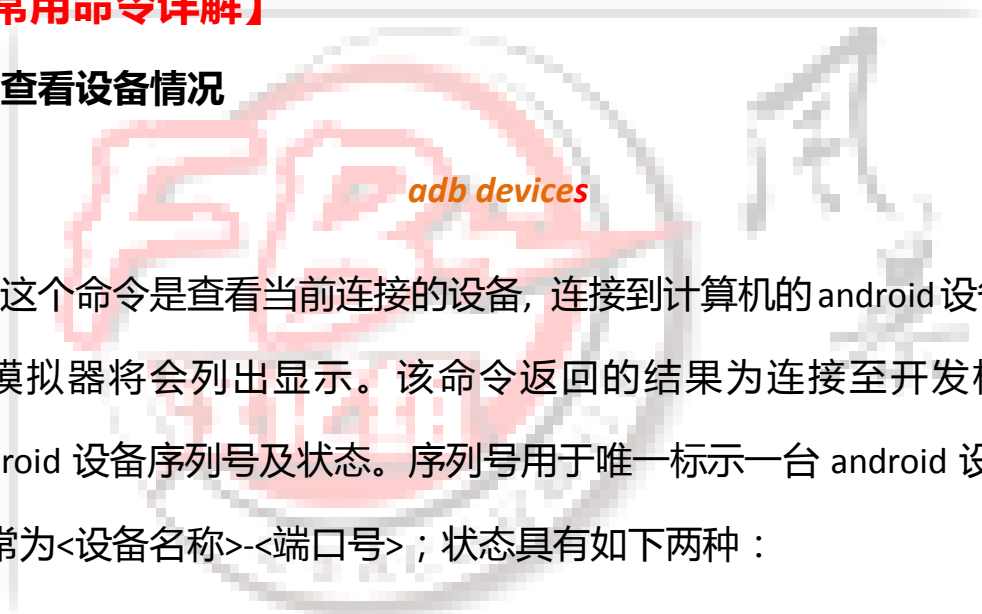
C:\Users\milk>adb
Android Debug Bridge version 1.0.20

-d                                - directs command to the on
```

至此，ADB 的安装就成功了，我们继续来看看 ADB 常用的命令。

【常用命令详解】

1、查看设备情况



这个命令是查看当前连接的设备，连接到计算机的 android 设备或者模拟器将会列出显示。该命令返回的结果为连接至开发机的 android 设备序列号及状态。序列号用于唯一标示一台 android 设备，通常为<设备名称>-<端口号>；状态具有如下两种：

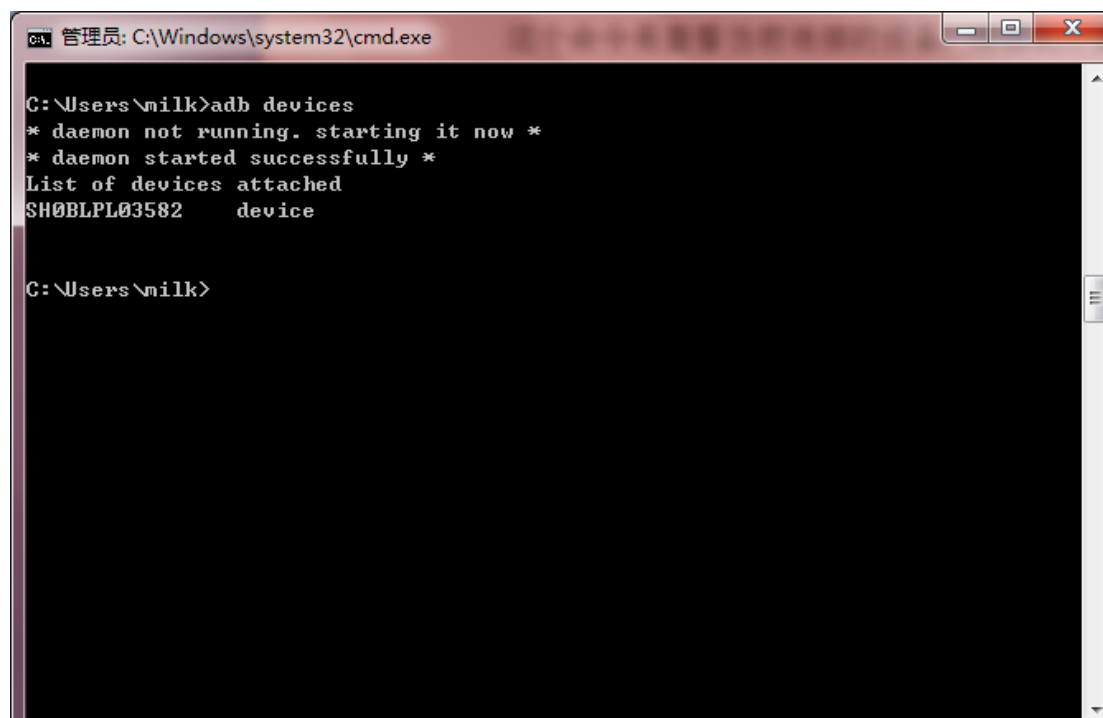
Offline——设备没有连接到开发机或无响应；

Device——设备已经连接到开发机。注意，该状态并不表示 android 设备可用，当 android 设备处于启动阶段时，若连接成功也会返回该状态。

如前所述 adb 首先会启动 server，该 server 使用 5037 端口进行监听，并尝试使用 5555 ~ 5585 端口连接 Android 设备/模拟器。当多台



android 设备连接至同一开发机时，需要使用-s <序列号> 区分设备进行操作。



```
C:\Users\milk>adb devices
* daemon not running. starting it now *
* daemon started successfully *
List of devices attached
SH0BLPL03582    device

C:\Users\milk>
```

2、安装软件

adb install <apk 文件路径>

这个命令将指定的 apk 文件安装到设备上。

如下图所示，我们可以先进入到软件所在的文件夹，这里在 C 盘下建立了个 adb 文件夹，里面有个 APK 名为 autokiller，我们使用 install 命令来操作：



```
管理员: C:\Windows\system32\cmd.exe
C:\adb>adb install autokiller.apk
919 KB/s (210977 bytes in 0.224s)
  pkg: /data/local/tmp/autokiller.apk
Success
C:\adb>
```

adb install -r

这句命令就是重新安装该软件的意思。

```
C:\adb>adb install -r autokiller.apk
1346 KB/s (210977 bytes in 0.153s)
  pkg: /data/local/tmp/autokiller.apk
Success
C:\adb>
```

3、卸载软件

adb uninstall <软件名>

adb uninstall -k <软件名>

如果加 -k 参数,为卸载软件但是保留配置和缓存文件.

4、进入设备或模拟器的 shell

adb shell

```
C:\adb>adb shell
# exit
exit
C:\adb>
```

通过上面的命令，就可以进入设备或模拟器的 shell 环境中，在这个 Linux Shell 中，你可以执行各种 Linux 的命令，另外如果只想执行一条 shell 命令，可以采用以下方式：

adb shell [command]

如：*adb shell dmesg* 就会打印出内核的调试信息。

5、发布端口

可以设置任意的端口号，做为主机向模拟器或设备的请求端口。

如：

adb forward tcp:5555 tcp:8000

6、从电脑上发送文件到设备

adb push <本地路径> <远程路径>

用 push 命令可以把本机电脑上的文件或者文件夹复制到设备(手机)，此处的本地就是指电脑，远程就是指手机或者模拟器。

```
C:\adb>adb push c:\adb\autokiller.apk /sdcard/
607 KB/s (210977 bytes in 0.339s)
C:\adb>
```

此处需要注意的是，电脑上的路径是“C:\”，手机上的是“/sdcard/”这样的，两处的斜杠方向不同。

7、从设备上下载文件到电脑

adb pull <远程路径> <本地路径>

用 pull 命令可以把设备(手机)上的文件或者文件夹复制到本机电脑中。因为与命令 6 类似，此处不再重复。

8、查看 bug 报告

adb bugreport

因为 bug 报告太多了，就不截图了，嘻嘻。

9、在 adb shell 下可查看手机系统的具体命令

- *getprop*: 查看机器的全部信息参数，在这能看到你机器的全部信息参数，从你的硬件信息到所刷的 ROM 版本信息。

```
C:\Users\milk>adb shell
adb server is out of date. killing...
* daemon started successfully *
$ getprop
getprop
[persist.sys.timezone]: [Asia/Shanghai]
[ro.secure]: [1]
[ro.allow.mock.location]: [0]
[ro.debuggable]: [1]
[persist.service.adb.enable]: [1]
[ro.factorytest]: [0]
```

- *getprop ro.serialno* 查看机器的 SN 号

```
$ getprop ro.serialno
getprop ro.serialno
SH0BLPL03582
$
```

从图中可以看出，其实 SN 号在 *getprop* 的命令中已经获取了，才命令指示 *getprop* 的一个子部分而已，下面的命令相同。

- `getprop ro.carrier` 查看机器的 CID 号
- `getprop ro.hardware` 查看机器板子代号，可以看到 G7 的开发代号名为 bravo!

```
$ getprop ro.hardware
getprop ro.hardware
bravo
$
```

- `getprop ro.bootloader` 查看 SPL (Hboot) 版本号

```
$ getprop ro.bootloader
getprop ro.bootloader
0.93.0001
$
```

10、ubuntu 使用时出现的问题

在使用 **ubuntu** 时发现如下问题：使用 adb devices 不能发现已经连接的 android 设备。产生该问题的原因在于 adb 命令首次运行时会启动一个 server 守护进程，用于与 android 设备的交互。由于 ubuntu 的权限问题导致该守护进程无法访问设备。因此可以以如下方式解决 (注意，需将 adb 设置入 root 环境变量或者使用全路径)：

`sudo adb start-server`

显示守护进程启动成功后再以 adb XXX 进行使用即可，不再使用 adb 时可以通过

`sudo adb kill-server`

结束守护进程。



11、使用 adb 进行 recovery 的过程

adb shell <command> - 让手机执行命令，<command>就是手机执行的命令。如：

adb shell flash_image recovery /sd-card/recovery-RAv1.0G.img 执行将 recovery-RAv1.0G.img 写入到 recovery 区中。

我们刷 recovery 时一般按下顺序执行：

```
adb shell mount -a
```

```
adb push recovery-RAv1.0G.img /system/recovery.img
```

```
adb push recovery-RAv1.0G.img /sdcard/recovery-RAv1.0G.img
```

```
adb shell flash_image recovery /sdcard/recovery-RAv1.0G.img reboot
```

【ADB 命令列表】

Adb 命令详表

```
C:\Users\milk>adb
Android Debug Bridge version 1.0.20
```

-d - directs command to the only connected USB device. returns an error if more than one USB device is present.

-e - directs command to the only running emulator, returns an error if more than one emulator is running.

-s <serial number> - directs command to the USB device or emulator with the given serial number

-p <product name or path> - simple product name like 'sooner', or a relative/absolute path to a product out directory like 'out/target/product/sooner'. If -p is not specified, the ANDROID_PRODUCT_OUT environment variable is used, which must be an absolute path.

devices - list all connected devices

device commands:

adb push <local> <remote> - copy file/dir to device

adb pull <remote> <local> - copy file/dir from device



adb sync [<directory>] - copy host->device only if changed
(see 'adb help all')

adb shell - run remote shell interactively

adb shell <command> - run remote shell command

adb emu <command> - run emulator console command

adb logcat [<filter-spec>] - View device log

adb forward <local> <remote> - forward socket connections
forward specs are one of:
tcp:<port>
localabstract:<unix domain socket name>
localreserved:<unix domain socket name>
localfilesystem:<unix domain socket name>
dev:<character device name>
jdwp:<process pid> (remote only)

adb jdwp - list PIDs of processes hosting a JDWP transport

adb install [-l] [-r] <file> - push this package file to the device and install it
('-l' means forward-lock the app)
('-r' means reinstall the app, keeping its data)

adb uninstall [-k] <package> - remove this app package from the device
('-k' means keep the data and cache directories)

adb bugreport - return all information from the device that should be included in a bug report.

adb help - show this help message

adb version - show version num

DATAOPTS:

(no option) - don't touch the data partition

-w - wipe the data partition

-d - flash the data partition

scripting:

adb wait-for-device - block until device is online

adb start-server - ensure that there is a server running

adb kill-server - kill the server if it is running

adb get-state - prints: offline | bootloader | device

adb get-serialno - prints: <serial-number>

adb status-window - continuously print device status for a specified device

adb remount - remounts the /system partition on the device read-write

adb root - restarts adb with root permissions

networking:

adb ppp <tty> [parameters] - Run PPP over USB.
Note: you should not automatically start a PDP connection.
<tty> refers to the tty for PPP stream. Eg. dev:/dev/omap_csmi_tty1
[parameters] - Eg. defaultroute debug dump local notty usepeerdns

adb sync notes:

adb sync [<directory>]



<localdir> can be interpreted in several ways:

- If <directory> is not specified, both /system and /data partitions will be updated.
- If it is "system" or "data", only the corresponding partition is updated.

Adb 命令部分中文表

下列表格列出了 adb 支持的部分命令，并对它们的意义和使用方法做了说明：

分类	命令	描述	说明
Options	-d	仅仅通过 USB 接口来管理 adb	如果不只是用 USB 接口来管理则返回错误
	-e	仅仅通过模拟器实例来管理 adb	如果不是只通过模拟器实例管理则返回错误
	-s <serialNumber> <序列号>	通过模拟器/设备的允许的命令号码来发送命令来管理 adb (比如: "emulator-5556")	如果没有指定号码，则会报错
General	devices	查看所有连接模拟器/设备的设施的清单	
	help	查看 adb 所支持的所有命令。	
	version	查看 adb 的版本序列号	
Debug	logcat [<option>] [<filter-specs>]	将日志数据输出到屏幕上	



	bugreport	查看 bug 的报告 如 dumphsys , dumpstate 和 logcat 信息。	
	jdwp	查看指定的设施的可用的 JDWP 信息	可以用 forward jdwp:<pid> 端口映 射信息来连接指定 的 JDWP 进程.例如 : adb forward tcp:8000 jdwp:472 jdb -attach localhost:8000
Data	install <path-to-apk>	安装 Android (可以模拟器/设 施的数据文件.apk 指定完整的路径)	
	pull <remote> <local>	将指定的文件从模拟器/设施 的拷贝到电脑上	
	push <local> < remote>	将指定的文件从电脑上拷贝到 模拟器/设备中	
Ports and Networking	forward <local > <remote>	用本地指定的端口通过 socket 方法远程连接模拟器/设施	
	ppp <tty> [par m]...	通过 USB 运行 ppp , 需要提醒 你的不能自动启动 PDP 连接	
Scripting	get-serialno	查看 adb 实例的序列号	
	get-state	查看模拟器/设施的当前状态	

		<p>你可以提前把命令转载在 adb 的命令器中,在命令器中的命令在模拟器/设备连接之前是不会执行其它命令的。示例如下:</p> <pre>adb wait-for-device shell getprop</pre> <p>需要提醒的是这些命令在所有的系统启动启动起来之前是不会启动 adb 的,所以在所有的系统启动起来之前你也不能执行其它的命令. 比如 运用 install 的时候就需要 Android 包,这些包只有系统完全启动时才安装。例如：</p> <pre>adb wait-for-device install <app>.apk</pre> <p>上面的命令只有连</p>
--	--	---



			接上了模拟器/设备 连接上了 adb 服务才 会被执行，而在 Android 系统完全启 动前执行就会有错 误发生
Server	start-server	选择服务是否启动 adb 服务进 程	
	kill-server	终止 adb 服务进程	
Shell	shell	通过远程 shell 命令来控制模拟 器/设备实例	
	shell [<shellCo mmand>]	连接模拟器/设施执行 shell 命 令，执行完毕后退出远程 shell 端	

风暴数码-兄弟连

bbs.fengbao.com

2011 年 1 月 1 日