

# Hacking Android bootloaders

Sachin Patil

Indian Institute of Technology, Bombay

isachin@iitb.ac.in

November 29, 2012



# Why?

## why do I need it?

- I want dual/multiboot phone
- testing
- for fun



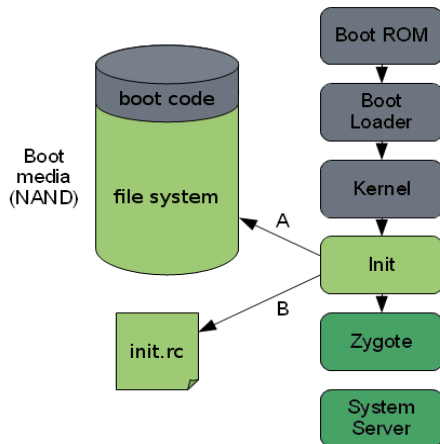
# Introduction

- Two different versions of Android
  - Ice cream sandwich - NAND
  - Jelly bean - SDcard
- Cyanogen(mod)



# Booting

## Booting Android...



# '/dev/mtd/' Partitions

dev:	size	erasesize	name
-----			
mtd0:	00500000	00020000	"recovery"
mtd1:	00500000	00020000	"boot"
mtd2:	00180000	00020000	"splash"
mtd3:	00080000	00020000	"misc"
mtd4:	02580000	00020000	"cache"
mtd5:	0dc00000	00020000	"system"
mtd6:	0a280000	00020000	"userdata"
mtd7:	01500000	00020000	"oem"
mtd8:	00180000	00020000	"persist"
mtd9:	00f00000	00020000	"fota"



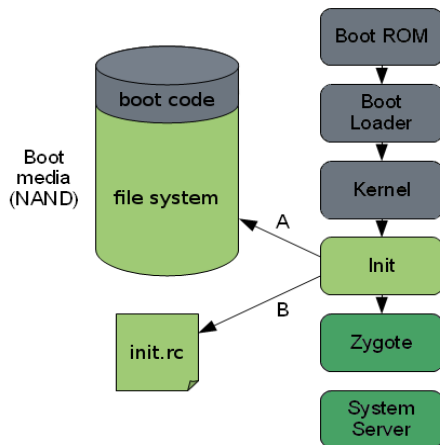
# inside 'boot.img'

tools/mkbootimg/bootimg.h

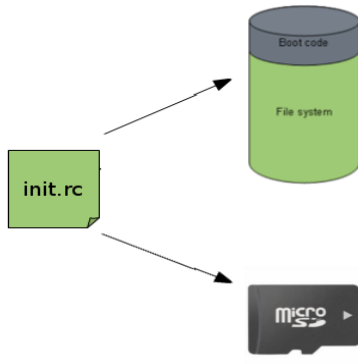
boot header	about 2k
kernel	kernel.gz
ramdisk	ramdisk.gz
second stage	(optional)



# Boot sequence



- Mounting
- Setting premissions
- Starting services





```
1 | ' /META-INF/com/google/android/updater-script '
2 | ' boot.img/ramdisk/init.rc '
```

```
1 | '/META-INF/com/google/android/updater-script'
```

```
2 | 'boot.img/ramdisk/init.rc'
```

```
1 | '/META-INF/com/google/android/updater-script'
```

```
2 | 'boot.img/ramdisk/init.rc'
```

## NAND

- `'mount yaffs2 mtd@system /system'`
- `'mount yaffs2 mtd@system /system ro remount'`
- `'mount yaffs2 mtd@userdata /data nosuid nodev'`

## SDcard

- `'mount ext4 /dev/block/mmcblk0p3 /system'`
- `'mount ext4 /dev/block/mmcblk0p3 /system ro remount'`
- `'mount ext4 /dev/block/mmcblk0p2 /data nosuid nodev'`

# updater-script

‘/META-INF/com/google/android/updater-script’

## mount

- `mount("yaffs2", "MTD", "system", "/system");`
- to**
- `run_program("/sbin/mount", "dev/block/mmcblk0p3",  
"/system");`



‘/META-INF/com/google/android/updater-script’

format

- ‘format("yaffs2", "MTD", "system", "0", "/system");’

to

- ‘run\_program("/sbin/mkfs.ext2", "dev/block/mmcblk0p3");’

# /sdcard/bootnand.sh

```
1  #!/bin/bash
2  # boot from NAND
3
4  # fill 'boot' partition with zero
5  cat /dev/zero > /dev/mtd/mtd1
6
7  # flash image
8  flash_image boot /sdcard/multiboot/boot.img
9  echo "NAND boot enabled"
10 reboot
```



# /sdcard/bootnand.sh

```
1  #!/bin/bash
2  # boot from NAND

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/boot.img
7  echo "NAND boot enabled"
8  reboot
```





# /sdcard/bootnand.sh

```
1  #!/bin/bash
2  # boot from NAND

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/boot.img
7  echo "NAND boot enabled"
8  reboot
```



# /sdcard/bootnand.sh

```
1  #!/bin/bash
2  # boot from NAND

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/boot.img
7  echo "NAND boot enabled"
8  reboot
```



# /sdcard/bootsd.sh

```
1  #!/bin/bash
   # boot from SDcard

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/bootsd.img
7  echo "SDcard boot enabled"
8  reboot
```



# /sdcard/bootsd.sh

```
1  #!/bin/bash
2  # boot from SDcard

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/bootsd.img
7  echo "SDcard boot enabled"
8  reboot
```



# /sdcard/bootsd.sh

```
1  #!/bin/bash
2  # boot from SDcard

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/bootsd.img
7  echo "SDcard boot enabled"
8  reboot
```



# /sdcard/bootsd.sh

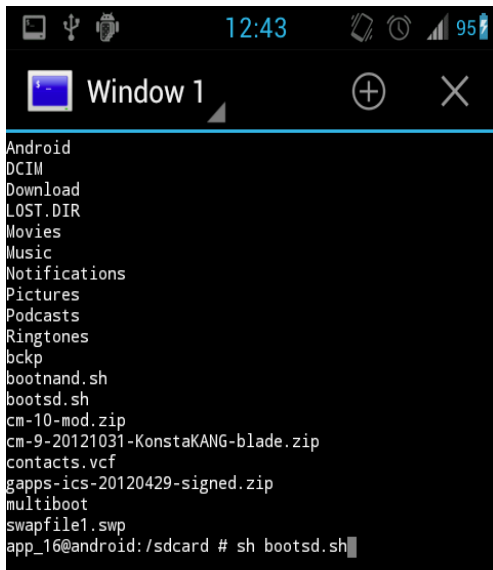
```
1  #!/bin/bash
2  # boot from SDcard

3  # fill 'boot' partition with zero
4  cat /dev/zero > /dev/mtd/mtd1

5  # flash image
6  flash_image boot /sdcard/multiboot/bootsd.img
7  echo "SDcard boot enabled"
8  reboot
```



# terminal emulator



The screenshot shows a terminal emulator window with a title bar that includes a window icon, the text 'Window 1', and standard window controls (minimize, maximize, close). The terminal content displays a directory listing of an Android device's storage, showing various folders and files. The prompt indicates the user is in the '/sdcard' directory and has executed the 'sh bootsd.sh' command.

```
Android
DCIM
Download
LOST.DIR
Movies
Music
Notifications
Pictures
Podcasts
Ringtones
bckp
bootnand.sh
bootsd.sh
cm-10-mod.zip
cm-9-20121031-KonstaKANG-blade.zip
contacts.vcf
gapps-ics-20120429-signed.zip
multiboot
swapfile1.swp
app_16@android:/sdcard # sh bootsd.sh
```



Demo





# References

## web links

- [elinux.org/Android\\_Bootimg](http://elinux.org/Android_Bootimg)
- [android.googlesource.com/platform/system/core/+/master/init/readme.txt](http://android.googlesource.com/platform/system/core/+/master/init/readme.txt)
- [github.com/android/platform\\_system\\_core/blob/master/mkbootimg/bootimg.h](http://github.com/android/platform_system_core/blob/master/mkbootimg/bootimg.h)
- <http://www.youtube.com/watch?v=NIBrSyxSYLo>

## forums

- [www.modaco.com/topic/356459-modtool-dual-boot-for-zte-blademodify-boot-tool-1408](http://www.modaco.com/topic/356459-modtool-dual-boot-for-zte-blademodify-boot-tool-1408)
- [forum.xda-developers.com/showthread.php?t=1823400](http://forum.xda-developers.com/showthread.php?t=1823400)



## downloads

- <https://github.com/psachin/Bootimg-scripts>
- <http://www.cyanogenmod.org/>
- <http://zteblade.eu/>
- [http://cyanogen-files.carneeki.net/flash\\_image.zip](http://cyanogen-files.carneeki.net/flash_image.zip)



Questions?

