평양 2407

Hacking North Korea’s Android

[@hackerfantastic] [@myhackerhouse] [https://hacker.house]

Introductions

Pyongyang 2407
Hacking North Korea

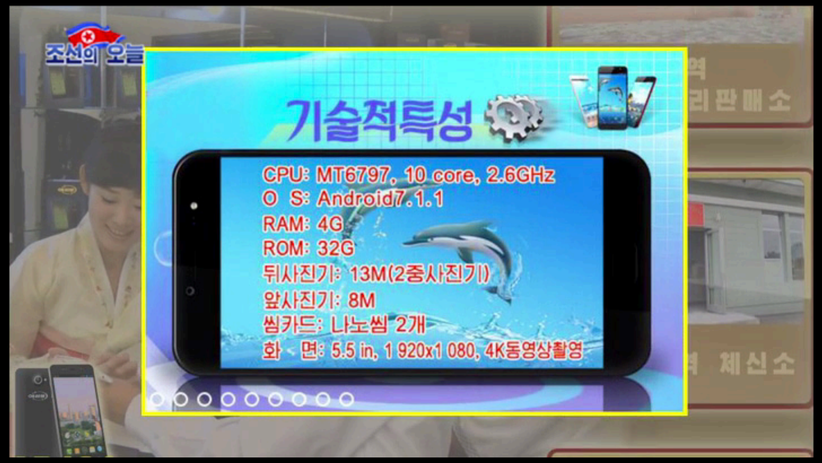
평양 2407 Smart Phone

… or surveillance tool? We explore, you decide!

- 34C3 - KCC release “평양 2407” disk image
- Locating and obtaining a phone
- Hardware teardown & booting disk image
- Android ROM reverse engineering
- Security features, user privacy & exploits/jailbreaking

평양 2407 Overview

koryolink™



North Korea smart phones

Pyongyang 2407
Hacking North Korea

“MediaTek” made in China

Pyongyang 2407
Hacking North Korea

평양 2407 Internals

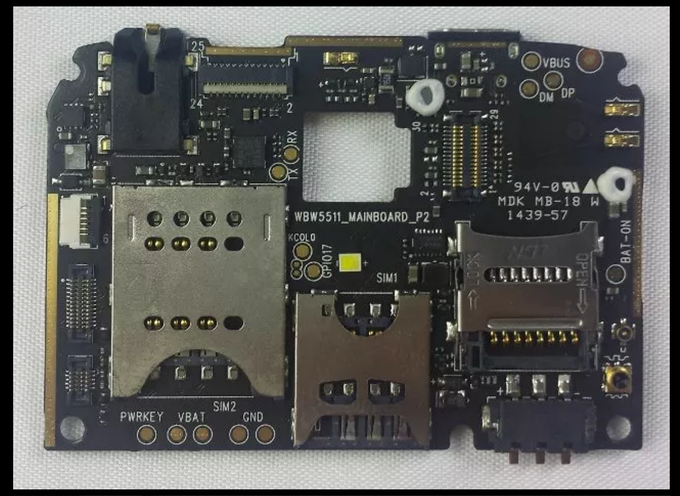
WBW5511_MAINBOARD_P2

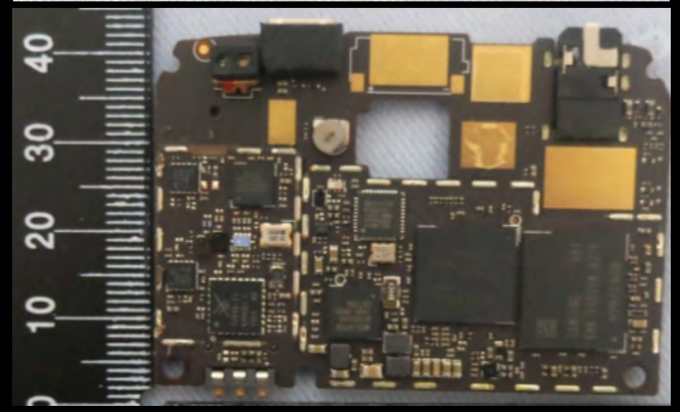
Mass-produced Mediatek device for overseas markets using Android, vendor deploys aftermarket Android with custom overlays & branding.

Several vendors supply identical WBW5511 based Android devices.

- **Gionee Ctrl V5 (India)**
- **Walton Primo H3 (Egypt / India)**
- **BLU Life Play 2 (USA)**

Exported to USA, certified in China for FCC https://fccid.io/YHLBLULIFEPLAY2 - ROMS are compatible between hardware devices.

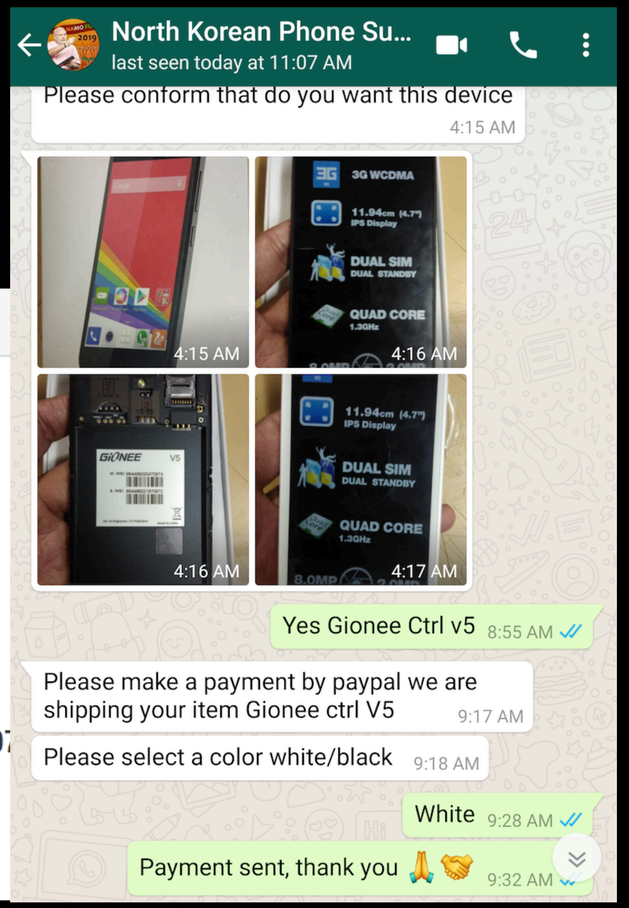




Google

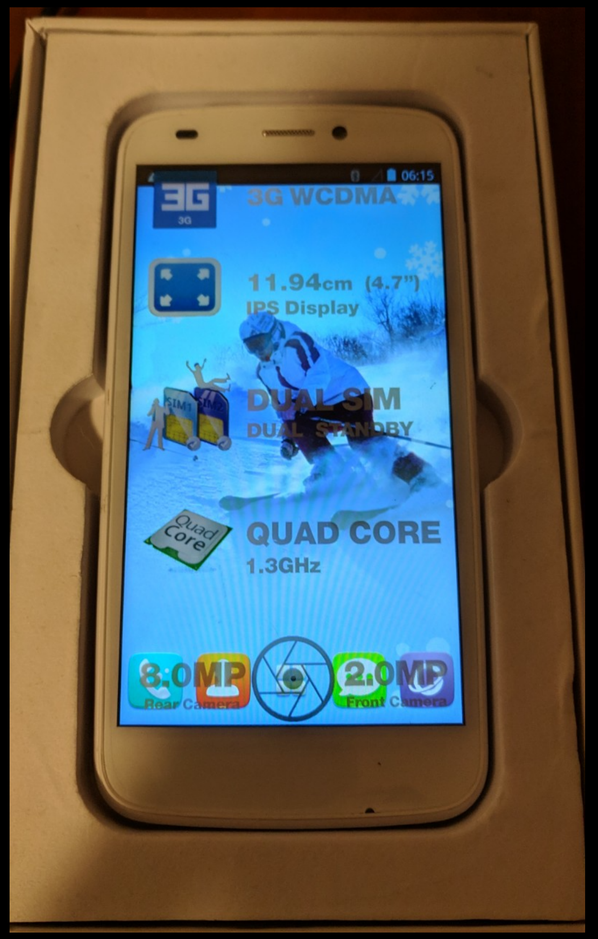
BLU Products, Inc. FCC Registration Number

FRN	**0019890946**
Registration Date:	06/02/2010 03:15:00 AM
Last Updated:	
Business Name:	CT Asia
Business Type:	Foreign Entity
Contact Organization:	
Contact Position:	Product Manager
Contact Name:	Mr Chris Chan
Contact Address:	Unit 01, 15/F, Seaview Centre, 139-141 Hoi Bun Road, Kwun Tong, Kowloon,Hong Kong9990
Contact Email:	chris@ctasiahk.com
ContactPhone:	852-27931198
ContactFax:	852-27931197



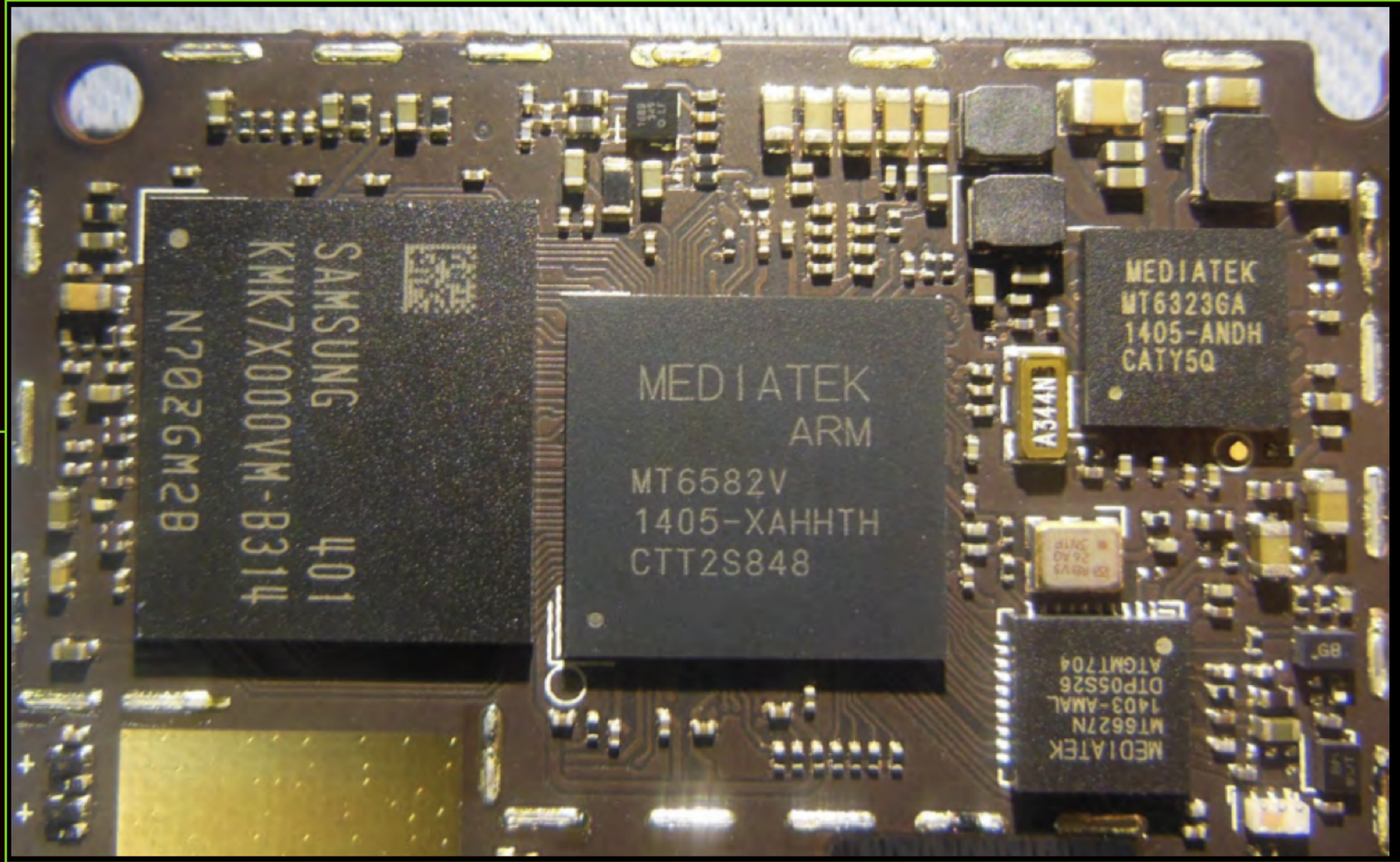
Where to buy phones?

Pyongyang 2407
Hacking North Korea



Booting 평양 2407 ROM

Pyongyang 2407
Hacking North Korea



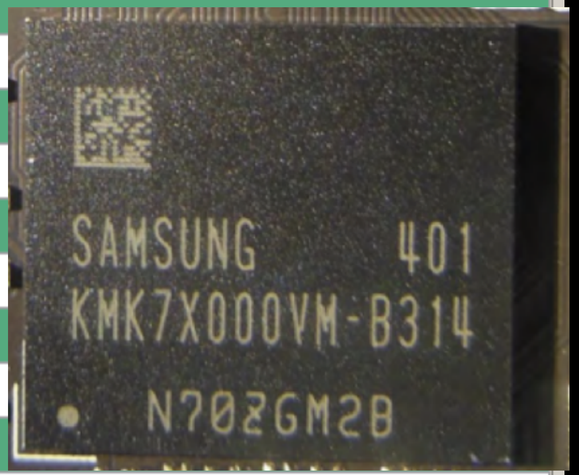
MT6582 is a highly integrated baseband platform incorporating modem, application processing and connectivity subsystems to enable 3G smart phone applications.

The chip integrates a **1.3GHz Quad-core ARM Cortex-A7,** an ARM Cortex-R4 MCU and a powerful multistandard video accelerator. The **MT6582 interfaces to NAND flash memory, LPDDR2 and LPDDR3** for optimal performance and also supports **booting from SLC NAND or eMMC** to minimize the overall BOM cost.

MEDIATEK MT6582 SoC

Format All + Download

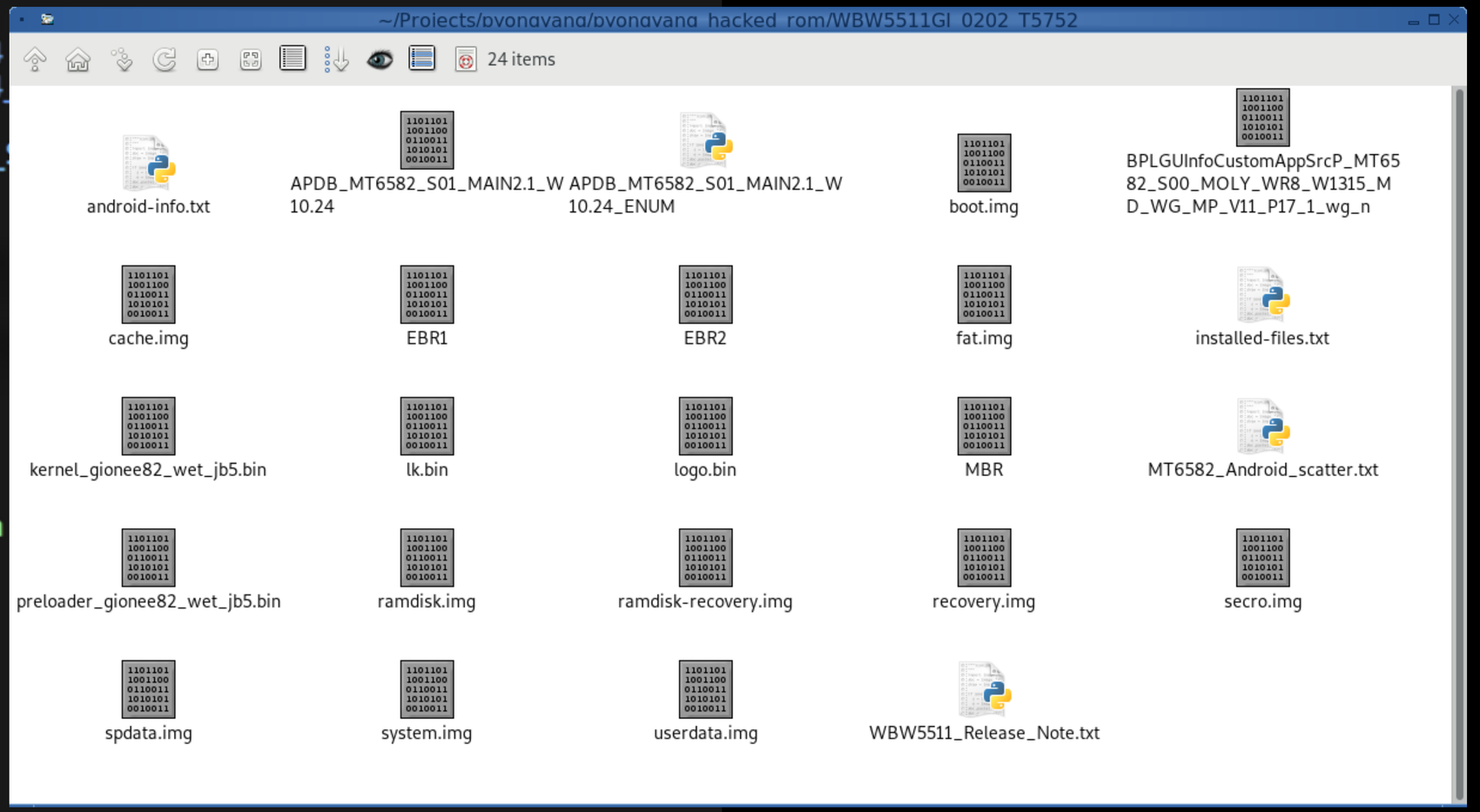
Name	Begin Address	End Address	Location
PRELOADER	0x0000000000000000	0x000000000001ba3b	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/preloader_gionee82_wet_jb5.bin
MBR	0x0000000000880000	0x00000000008801ff	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/MBR
EBR1	0x0000000000900000	0x00000000009001ff	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/EBR1
UBOOT	0x00000000025a0000	0x00000000025dfcaf	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/lk.bin
BOOTIMG	0x0000000002600000	0x0000000002a78fff	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/boot.img
RECOVERY	0x0000000003600000	0x0000000003af77ff	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/recovery.img
SEC_RO	0x0000000004600000	0x0000000004620fff	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/secro.img
LOGO	0x00000000046c0000	0x00000000047054f5	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/logo.bin
EBR2	0x00000000049c0000	0x00000000049c01ff	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/EBR2
SPDATA	0x0000000005440000	0x000000000586e093	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/spdata.img
ANDROID	0x0000000005c40000	0x0000000043456c77	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/system.img
CACHE	0x0000000065c40000	0x000000006643a0e7	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/cache.img
USRDATA	0x000000007d440000	0x000000007f3cf297	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/userdata.img
FAT	0x00000000f1440000	0x00000000f3d6507b	/home/fantastic/Projects/pyongyang/pyongyang_hacked_rom/WBW5511GI_0202_T5752/fat.img



Flash Format Layout

Pyongyang 2407
Hacking North Korea

```
android-info.txt
APDB_MT6582_S01_MAIN2.1_W10.24
APDB_MT6582_S01_MAIN2.1_W10.24_
boot.img
BPLGUInfoCustomAppSrcP_MT6582_
cache.img
EBR1
EBR2
fat.img
installed-files.txt
kernel_gionee82_wet_jb5.bin
lk.bin
logo.bin
MBR
MT6582_Android_scatter.txt
preloader_gionee82_wet_jb5.bin
ramdisk.img
ramdisk-recovery.img
recovery.img
secro.img
spdata.img
system.img
userdata.img
WBW5511_Release_Note.txt
```

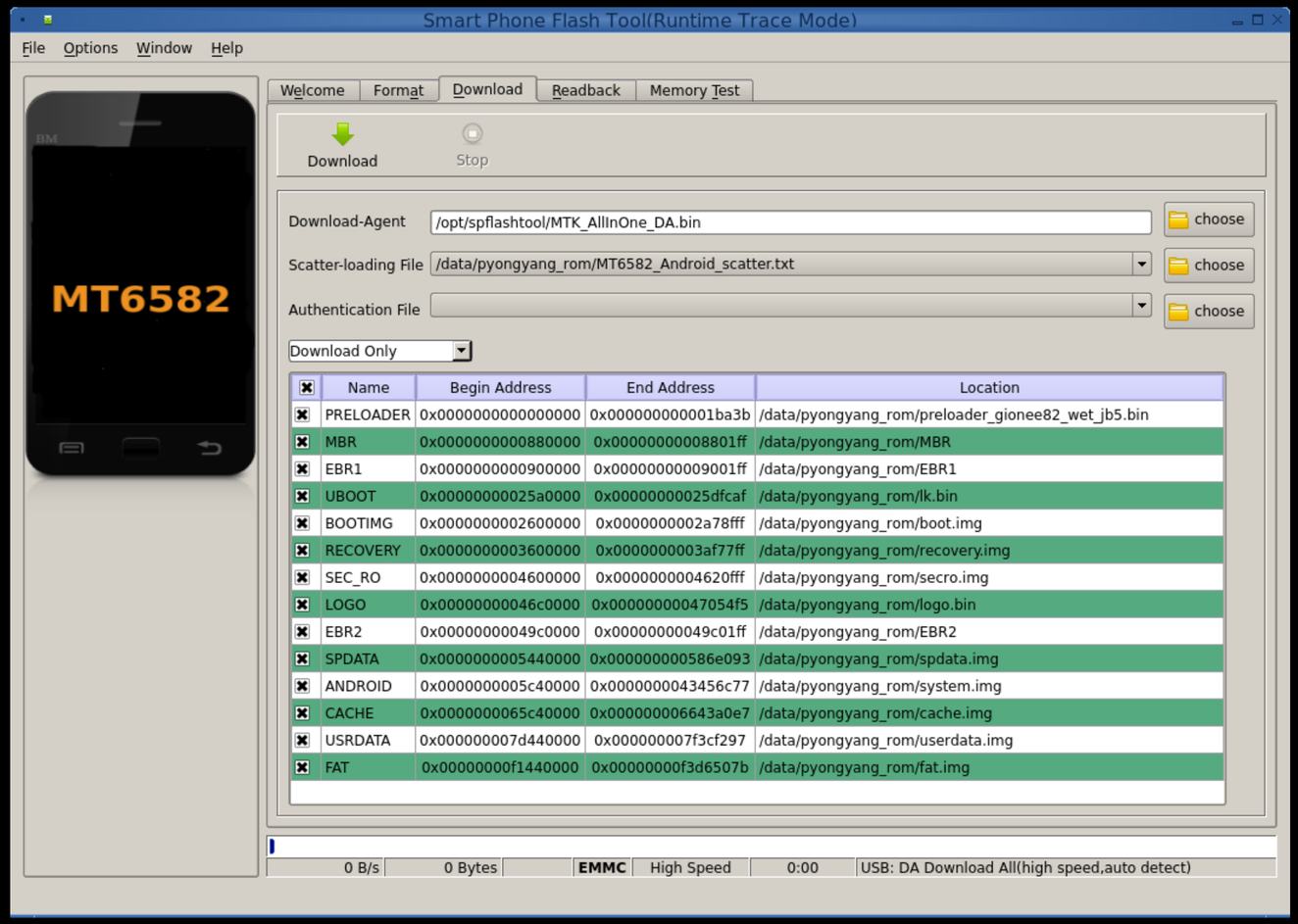


ROM WBW5511GI_0202_T5752

Pyongyang 2407
Hacking North Korea

```
==================
Build Notes
==================
1.版本目的:
验证前置群辉摄像头偏黄
ESD问题
2.修改的重大问题点:
3.测试重点:
4.应用数据库是否改变:

==================
Release Location
==================
FTP://18.8.8.2/software_release/WBW5511/WBW5511GI_0202_T5752.zip

==================
CR List
==================
[Repository Url]: http://192.168.110.97/svn/android_mtk_jb5_6582_wet/branches/branch_oversea_wbw5506_5511_5508_t5531_rel
CR01031315      Opened  WBW5511A        zhanghe 5511 sub camera 代码合入        svn revision(8991)
CR01227613      Opened  WBW5511GS       changlei        5511GS资源配置及菜单摆放        svn revision(8992,8990)
Tag url: http://192.168.110.97/svn/android_mtk_jb5_6582_wet/tags/TAG_WBW5511_T5752
Previous tag: TAG_WBW5511_T5751
Code difference: svn diff -r 8992:8987 http://192.168.110.97/svn/android_mtk_jb5_6582_wet/branches/branch_oversea_wbw5506_5511_5508_t5531_rel

[Repository Url]: http://192.168.110.97/svn/gionee_packages_apk_amigo/branches/branch_oversea_rom4.2.4_rel
CR01245664      Opened  WBW5885GI(Amigo)        liuliang        [5885GI_amigo]印度反馈:随变->壁纸中有Lockscreen和Menu wallpaper两项，这两项在navi l
auncher项目中无作用，应去掉  svn revision(29171)
Tag url: http://192.168.110.97/svn/gionee_packages_apk_amigo/tags/TAG_WBW5511_T5752
Previous tag: TAG_WBW5511_T5751
Code difference: svn diff -r 29171:29025 http://192.168.110.97/svn/gionee_packages_apk_amigo/branches/branch_oversea_rom4.2.4_rel
```

Release_Note FTP Info Leak




```
[134315.811873] usb 1-5: new high-speed USB device number 40 using xhci_hcd
[134315.954481] usb 1-5: New USB device found, idVendor=0e8d, idProduct=2000, bcdDevice= 1.00
[134315.954488] usb 1-5: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[134315.954492] usb 1-5: Product: MT65xx Preloader
[134315.954496] usb 1-5: Manufacturer: MediaTek
[134316.018783] cdc_acm 1-5:1.1: ttyACM0: USB ACM device
[134317.649985] usb 1-5: USB disconnect, device number 40
```

MediaTek provides tools and support for their overseas & mass-produced boards/devices.

Plugging in the device without a battery will briefly show a USB device…

“spflashtool” is MediaTek way of flashing any MT6582 based device ..

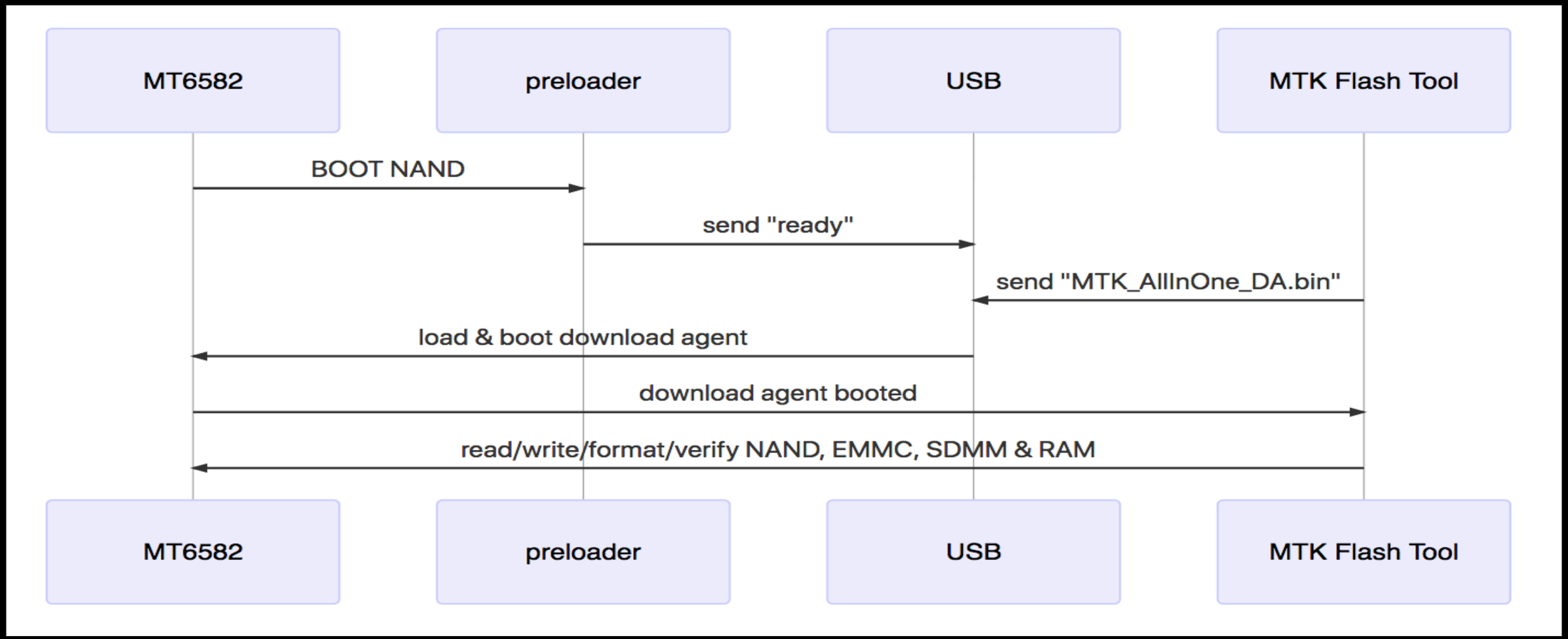
It works early in preloader boot process and allows raw memory & storage hardware access …

A binary “agent” is transmitted to the device to run in RAM.

MediaTek SP Flash Tool

Pyongyang 2407
Hacking North Korea

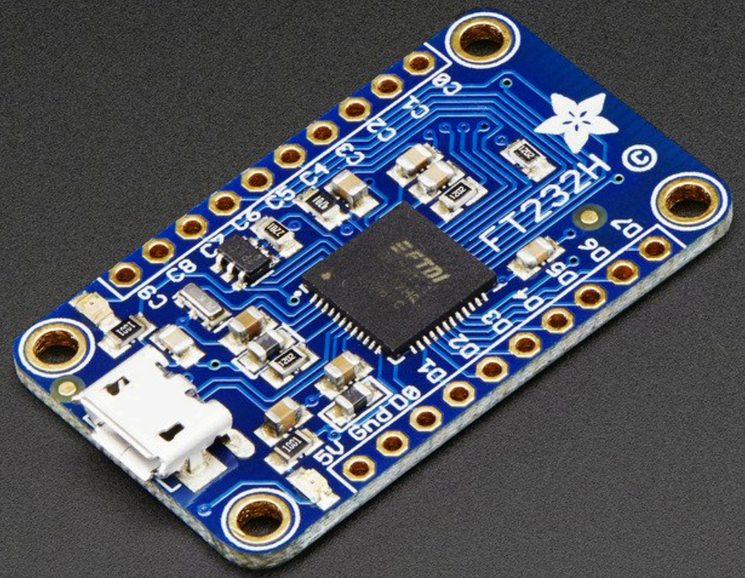
[@hackerfantastic] [@myhackerhouse] [https://hacker.house]

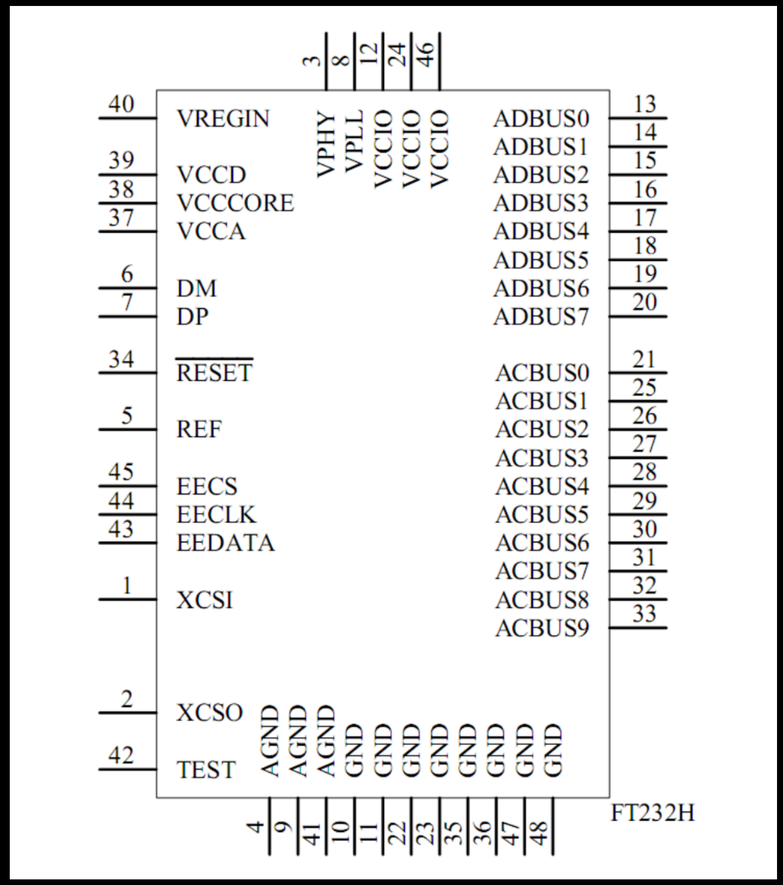


MT6582 download agent

```
144         bzero(&(sa.sin_zero),8);
145     printf("[ connecting to %s %d/tcp\n",server,port);
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161     host = gethostbyname(server);
162
163
164
165         exit(0);
166     }
```

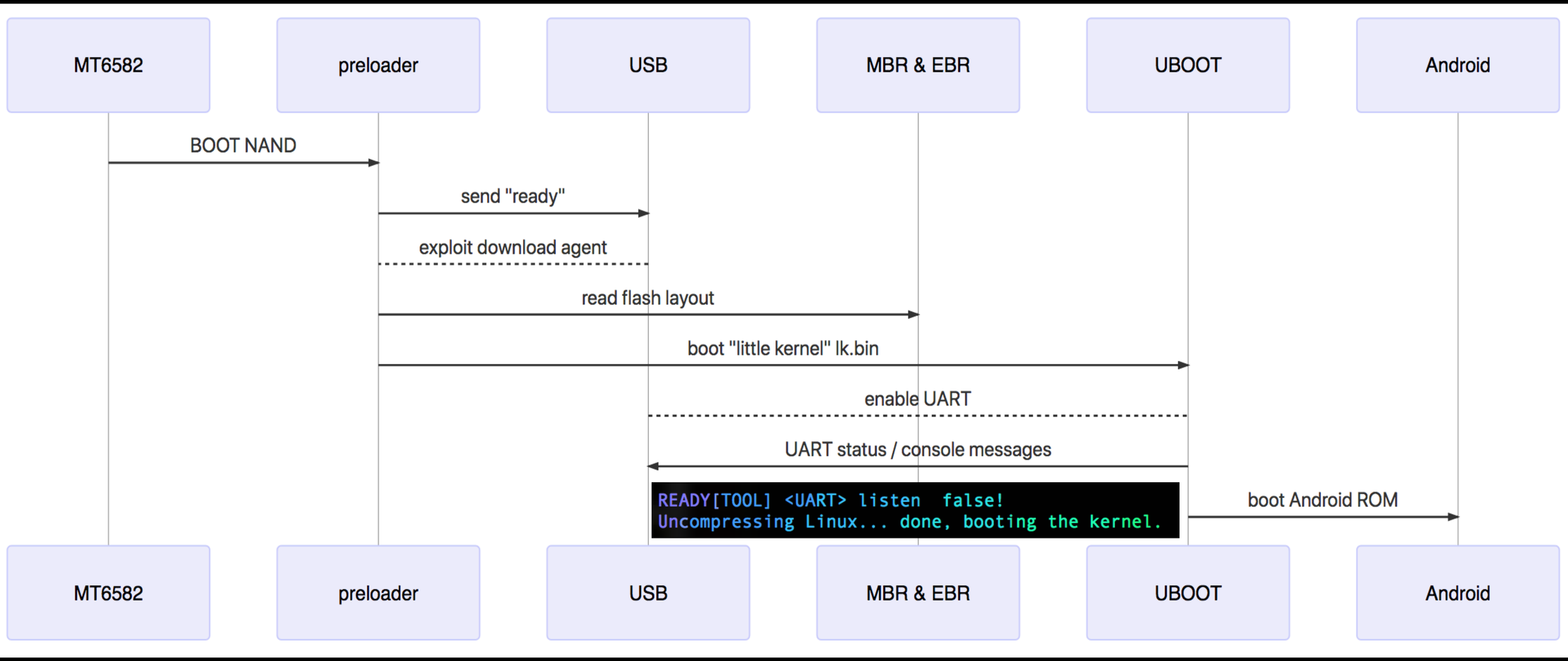
FT232H used for UART access
GND, D0, D1 wired to a MicroUSB cable to phone





WBW5511 UART cable

Pyongyang 2407
Hacking North Korea



MT6582 BOOT & USB access

Pyongyang 2407
Hacking North Korea

```
shell@평양:/system/bin # cat /proc/cmdline
console=ttyMT3,921600n1 vmalloc=496M slub_max_order=0 lcm=1-gn_tm_otm9605a fps=5911 bootprof.pl_t=3264 bootpr
of.lk_t=1621 printk.disable_uart=1 boot_reason=4 boot_mode=0
```

patch lk.bin, changes bootarg ok. no shell :(

```
1|shell@평양:/proc # cat cmdline
console=ttyMT3,921600n1 vmalloc=496M slub_max_order=0 lcm=1-gn_tm_otm9605a fps=5911 bootprof.pl_t=3990 bootpr
of.lk_t=1604 printk.disable_uart=0 boot_reason=4 boot_mode=0
```

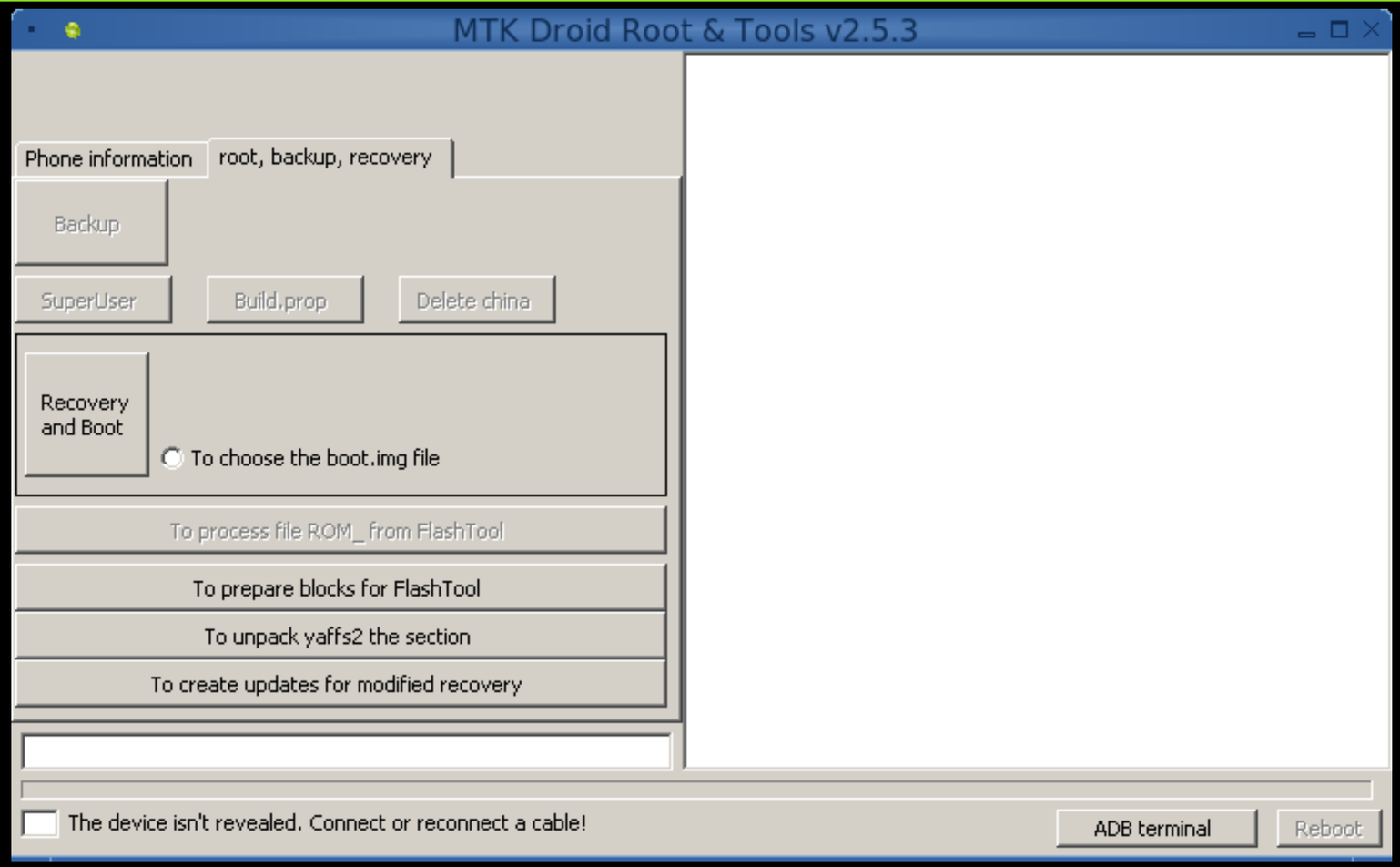
```
READY[TOOL] <UART> listen  false!
Uncompressing Linux... done, booting the kernel.
```

```
shell@평양:/proc # ls -al /dev/ttyMT*
crw------- root     root      204, 209 2014-01-01 09:47 ttyMT0
crw------- root     root      204, 210 2014-01-01 09:47 ttyMT1
crw-rw---- system   system    204, 211 2014-01-01 09:47 ttyMT2
crw------- root     root      204, 212 2014-01-01 09:47 ttyMT3
shell@평양:/proc # busybox microcom -s 921600 /dev/ttyMT0
shell@평양:/proc # busybox microcom -s 921600 /dev/ttyMT1
shell@평양:/proc # busybox microcom -s 921600 /dev/ttyMT2
shell@평양:/proc # busybox microcom -s 921600 /dev/ttyMT3
shell@평양:/proc #
```

MEDIATEK

lk.bin printk.disable_uart=0

Pyongyang 2407
Hacking North Korea



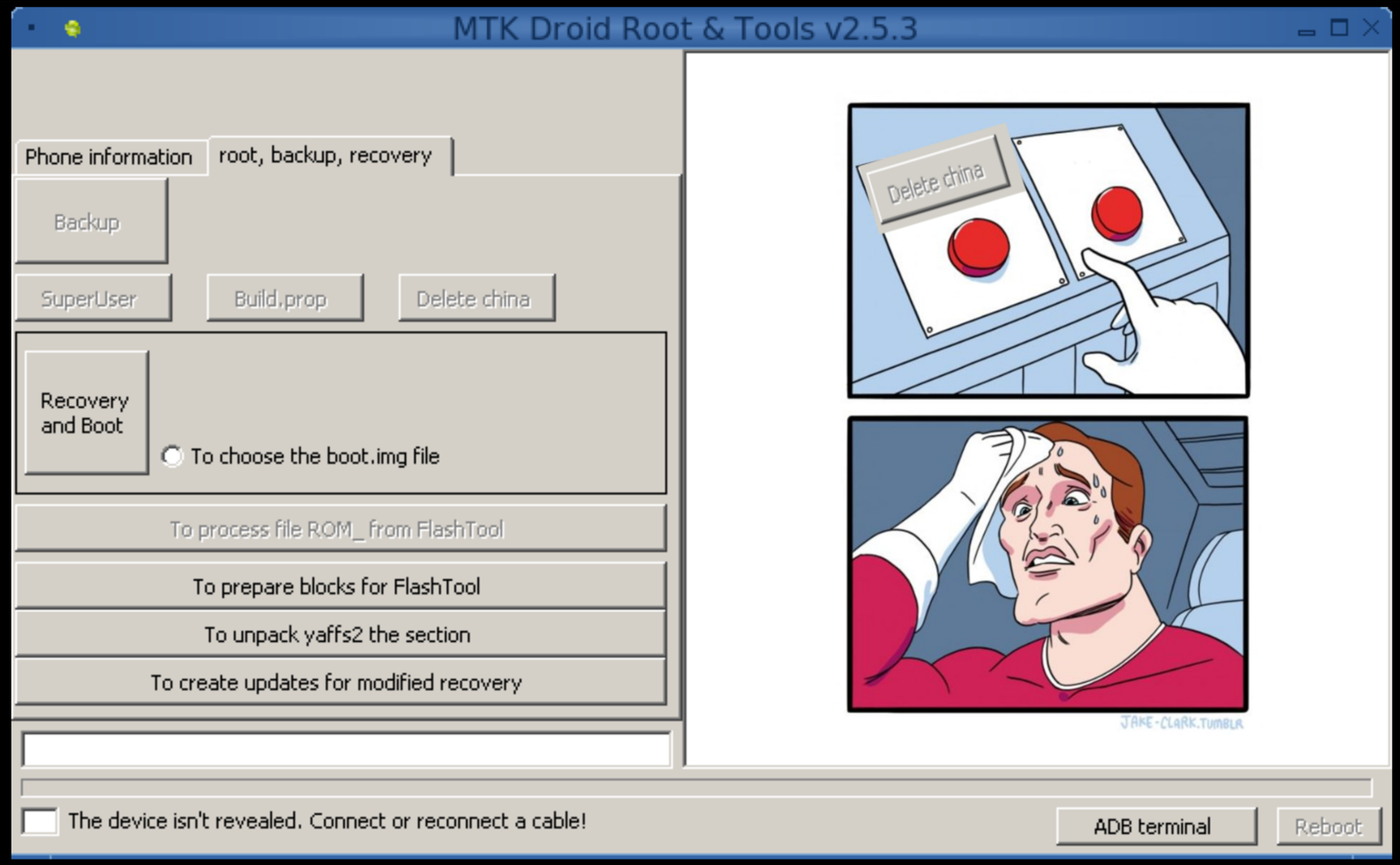
Windows binary executable

Can create “scatter file” backups, uses adb and custom recovery.img

Backups compatible with “spflashtool”

Back up and protect the preloader (no preloader, no RAM recovery)

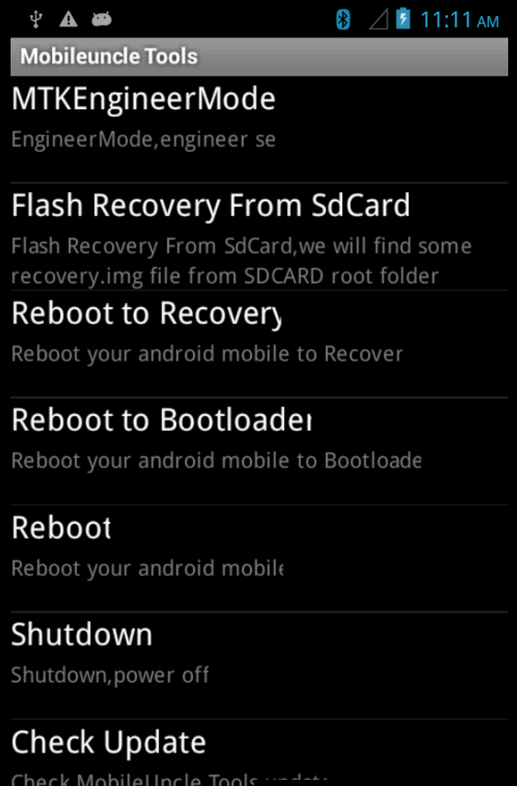
MTK Droid Root & Tools



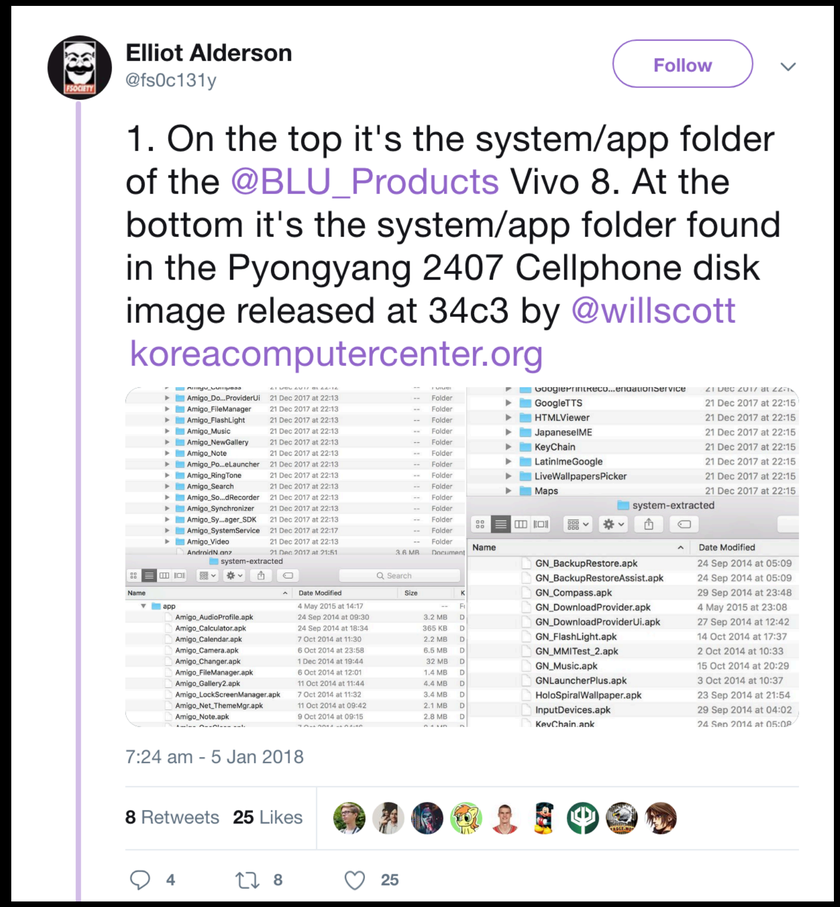
"Delete not used and Chinese apps. Reads from files_for_delete.txt"

A CWM backup & recovery from recovery.img mode.

Deleting China?




```
fs0c13ty # adb push recovery.img /sdcard/recovery.img
recovery.img: 1 file pushed. 7.4 MB/s (5208064 bytes in 0.672s)
```



Mobileuncle Tools: hi friend.

Pyongyang 2407
Hacking North Korea

평양 2407 Android

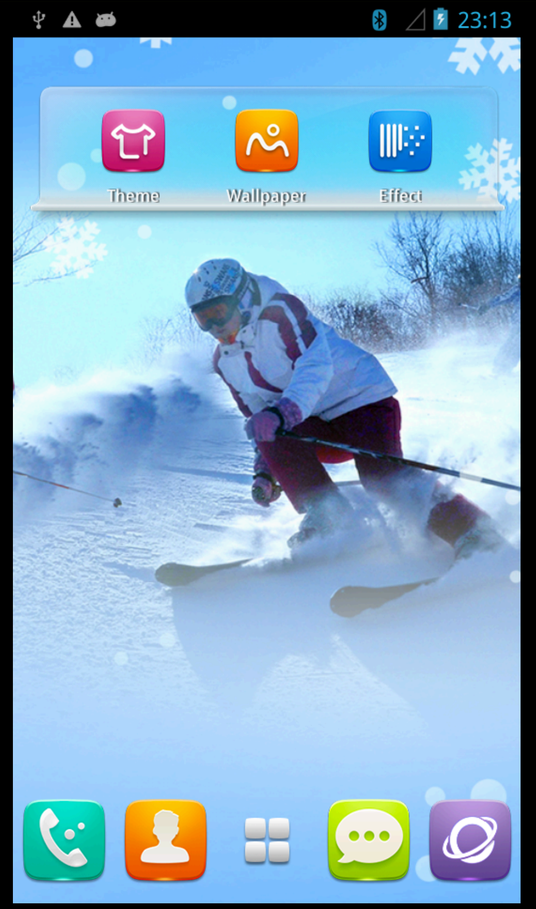
Based on Jellybean 4.2.2*

The Pyongyang 2407 Cellphone disk image contains the “system” and “data” ext4 partitions as tar files. It also contains a kernel zImage and ramdisk.gz.

Indicators from the device were removed and cleaned off before taken out of North Korea. Patching needed to work on aftermarket devices, added to github.

Patched ROM files, archive & tools https://github.com/hackerhouse-opensource/pyongyang_2407

More APK’s to install and play with on KCC http://www.koreacomputercenter.org/



DPRK Android Applications

Pyongyang 2407

Hacking North Korea








https://github.com/hackerhouse-opensource/pyongyang_2407/tree/master/wallpapers

North Korea Wallpapers

Pyongyang 2407
Hacking North Korea

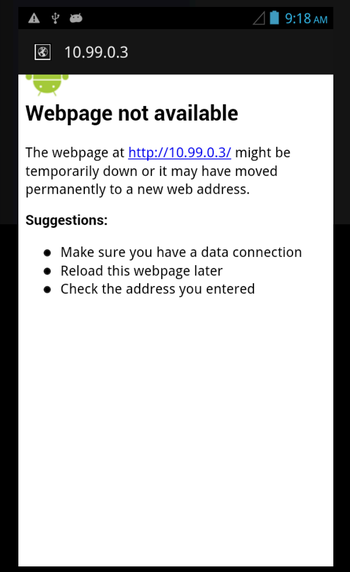
```
GET / HTTP/1.1
Host: 127.0.0.1
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
x-wap-profile: http://218.249.47.94/Xianghe/MTK_Phone_JB_UAprofile.xml
X-Requested-With: com.android.browser
User-Agent: Mozilla/5.0 (Linux; U; Android 4.2.2; en-us; 2407 Build/JDQ39) AppleWebKit/534.30 (KHTML, like Ge
cko) Version/4.0 Mobile Safari/534.30
Accept-Encoding: gzip,deflate
Accept-Language: en-US
Accept-Charset: utf-8, utf-16, *;q=0.7
```

Android Webview Browser locked to the KwangMyong intranet

Wifi disabled in ROM (attempted to enable, missing code)

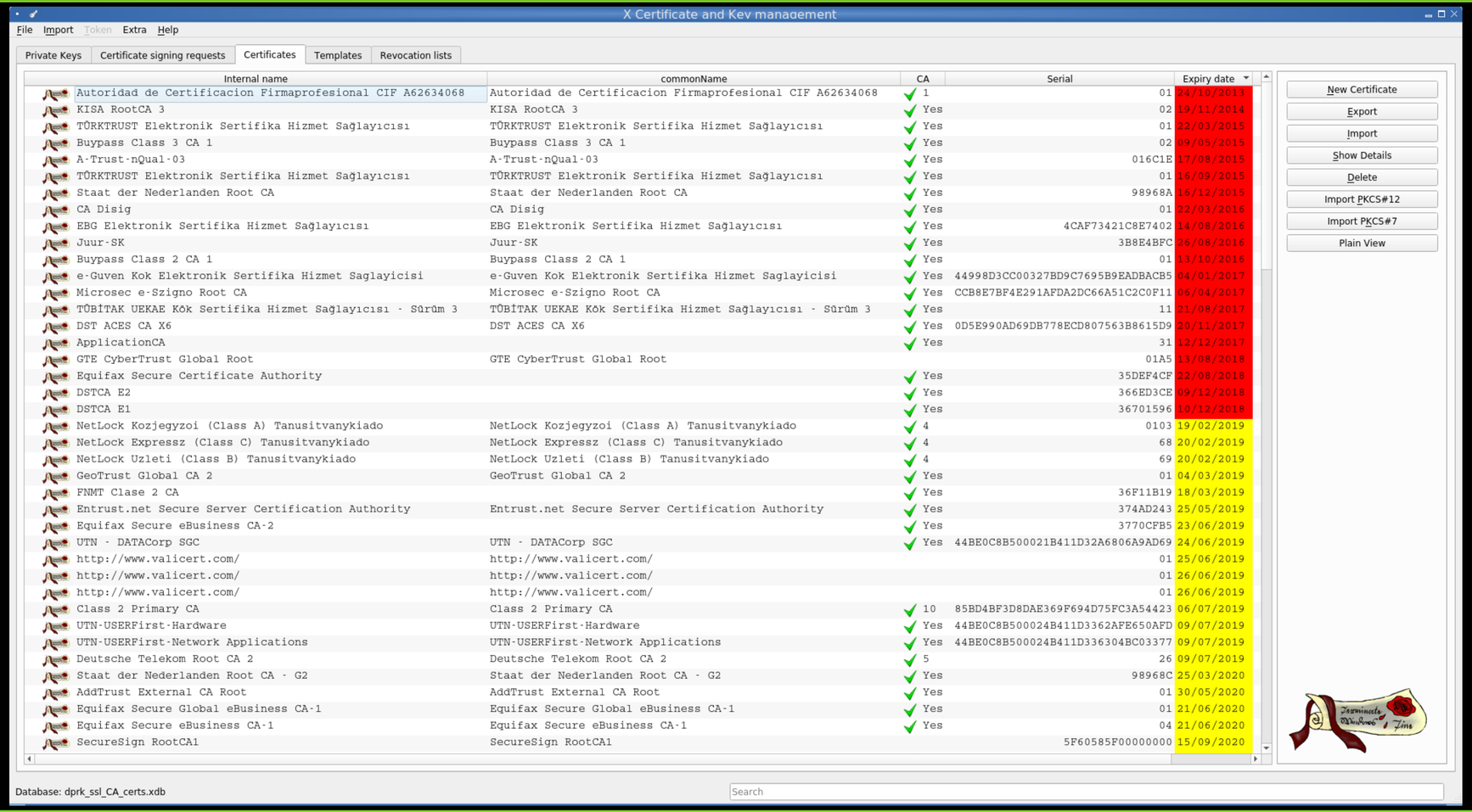
3G/CDMA Koroyolink network support

rtsp:// handler opens the com.gionee.video player activity



Web Browser: KwangMyong

Pyongyang 2407
Hacking North Korea



No DPRK root CA?

No Pyongyang root CA?

Test Certificate for Gionee.

Common global CA chain for Android WebView based browsers.

SSL Certificates

Pyongyang 2407
Hacking North Korea

```
144         bzero(&(sa.sin_zero),8);
145     printf("[ connecting to %s %d/tcp\n",server,port);
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161     host = gethostbyname(server);
162
163
164
165         exit(0);
166     }
```

package	code
com.android.providers.calendar	225
com.android.settings	4636
com.mediatek.bluetooth	87410
	2872714
	2872713
	2872710
com.mediatek.engineermode	3646633

Phone Secret Codes

North Korea has been described as a "*massive police state*", and its people "*under constant surveillance*".

booted mobile device, use it for a few days as a phone.

GOAL: watch "The Interview" movie without detection or alerting in the handset.

North Korean Surveillance

Pyongyang 2407
Hacking North Korea

Legal Files (adb sideload apps)

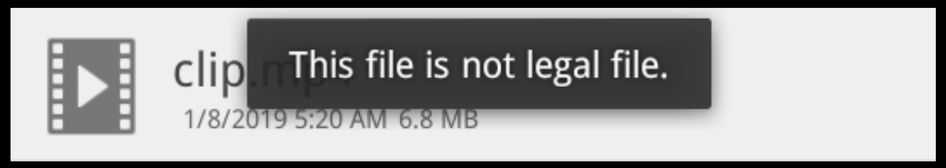
Pyongyang 2407
Hacking North Korea

```
Package: kuts.ktr.RFService
  Application Label: 전자서명조작체계 2.0
  Process Name: kuts.ktr.RFService
  Version: 2.0
  Data Directory: /data/data/
kuts.ktr.RFService
  APK Path: /system/app/RedFlag.apk
  UID: 10038
  GID: [1015, 1023, 1028]
  Shared Libraries: null
  Shared User ID: null

Package: kuts.ktr.RFService
  kuts.ktr.RFService.RFReceiver
    Permission: null
```

```
Uses Permissions:
  - android.permission.MOUNT_UNMOUNT_FILESYSTEMS
  - android.permission.WRITE_EXTERNAL_STORAGE
  - android.permission.RECEIVE_BOOT_COMPLETED
  - android.permission.WRITE_MEDIA_STORAGE
  - android.permission.REBOOT
  - android.permission.DEVICE_POWER
  - android.permission.READ_FRAME_BUFFER
  - android.permission.WAKE_LOCK
  - android.permission.VIBRATE
  - android.permission.GET_TASKS
  - android.permission.READ_PHONE_STATE
  - android.permission.WRITE_SETTINGS
  - android.permission.WRITE_SECURE_SETTINGS
  - android.permission.WRITE_SYNC_SETTINGS
  - android.permission.READ_SYNC_SETTINGS
  - android.permission.READ_EXTERNAL_STORAGE
Defines Permissions:
  - None
```

전자서명조작체계 2.0 (RedFlag)

```
D/gov_sign( 1866): --------.3gp------4----
D/gov_sign( 1866): get_flength : file name = /storage/sdcard0/DCIM/Camera/VID_20140107_031533.3gp, fSize = 35604822, POSTFIX_LEN = 0
D/gov_sign( 1866): MnsNative isNatSignFile : file name = /storage/sdcard0/DCIM/Camera/VID_20140107_031533.3gp, result = 264
E/MediaSelfSign( 1866): Call isSelfSignFile()-------------- successed!
E/MediaSelfSign( 1866): MSSMediaVerifyFileByPath()---------! return 0
E/MediaSelfSign( 1866): MSSMediaVerifyFileByPath()---------! return 0
E/MediaSelfSign( 1866): isSelfSignFile()---------successed! Err = 0, return 1
E/MediaSelfSign( 1866): Call CheckSelfSignFile()-------------- successed!
E/MediaSelfSign( 1866): MSSMediaVerifyFileByPath()---------! return 0
E/MediaSelfSign( 1866): CheckSelfSignFile()---------successed! Err = 0, return 2


D/gov_sign( 1866): --------.pdf------4----
D/gov_sign( 1866): get_flength : file name = /storage/sdcard0/프로그람자료/.KwangMyong/도움말.pdf, fSize = 1224818, POSTFIX_LEN = 0
D/gov_sign( 1866): MnsNative isNatSignFile : file name = /storage/sdcard0/프로그람자료/.KwangMyong/도움말.pdf, result = 1
D/SignalClusterView(  719): apply() setImageResource(mMobileTypeId) mShowSimIndicator = false
```



평양 *2407 has two kinds of signed content*

- **Media Self-Signed (Camera, Video, Docs)**
- **Nat Signed (gov_sign) approved gov media**

전자서명조작체계 logcat API trace info leak

NATISIGN & SELFSIGN

Pyongyang 2407
Hacking North Korea

```
1|shell@평양:/system/lib # ls -al *Legal*
-rw-r--r-- root     root         18020 2014-12-29 12:39 libLegalInterface.so
shell@평양:/system/lib # ls -al *sign*
-rw-r--r-- root     root         26244 2014-03-22 01:36 libmedianatsign.so
-rw-r--r-- root     root         61564 2015-04-23 05:45 libmediaselfsign.so
```

```
int MSSIsSelfSigned(int arg0, int arg1) {
    r1 = arg1;
    sp = sp - 0x30;
    r7 = arg0 + 0x0;
    r6 = r1 + 0x0;
    var_1C = **0xaa7c;
    *r1 = 0x0;
    r3 = lseek64(sign_extend_32(*(int16_t *)(arg0 + 0xe)), r1, 0x0) + 0x0;
    if (r1 == 0x0) {
            r0 = 0x0;
            if (r3 > 0x7) {
                    r0 = sign_extend_32(*(int16_t *)(r7 + 0xe));
                    r0 = lseek64(r0, r1, 0xfffffff8);
                    r0 = r0 + 0x1;
                    if (r0 != 0x0) {
                            r0 = sign_extend_32(*(int16_t *)(r7 + 0xe));
                            r7 = sp + 0x8;
                            if (read(r0, r7 + 0x0, 0x8) != 0x8) {
                                    r0 = 0x0;
                                    *r6 = 0x2a1;
                            }
                            else {
                                    r3 = memcmp(r7 + 0x0, "SELFSIGN", 0x8) + 0x0;
                                    r0 = 0x0 - r3;
                                    r0 = r0 + r3 + CARRY(FLAGS);
                            }
                    }
```

```
loc_3b9e:
    var_C = lseek64(sign_extend_32(*(int16_t *)(r5 + 0xe)), 0xe, 0x0);
    var_8 = 0xe;
    __android_log_print();
    if (var_8 != 0x0) {
            r2 = var_C + 0xfffffff8;
            lseek64(sign_extend_32(*(int16_t *)(r5 + 0xe)) + 0x0, 0x0, r2);
            read(sign_extend_32(*(int16_t *)(r5 + 0xe)), r6 + 0x0, 0x8);
            r0 = strncmp(r6 + 0x0, "NATISIGN" + 0x0, 0x8);
            if (r0 != 0x0) {
                    r2 = 0xfffffffc + var_C;
                    lseek64(sign_extend_32(*(int16_t *)(r5 + 0xe)), 0x0, r2);
                    if (read(sign_extend_32(*(int16_t *)(r5 + 0xe)), sp + 0x334, 0x4) == 0x4) {
                            var_4 = arg_310;
                            r0 = var_C - var_4;
                            r1 = var_8 - 0x0 + !CARRY(FLAGS);
                            var_8 = r1;
                            if ((r1 == 0x0) && (r0 < 0xdc << 0x1)) {
                                    r2 = var_4 + 0xfffffff8;
                                    lseek64(sign_extend_32(*(int16_t *)(r5 + 0xe)), 0x0, r2);
                                    read(sign_extend_32(*(int16_t *)(r5 + 0xe)), r6 + 0x0, 0x8);
                                    if (strncmp(r6 + 0x0, "NATISIGN" + 0x0, 0x8) != 0x0) {
                                            fclose(r5 + 0x0);
                                            r0 = 0x83 << 0x1;
                                    }
```

평양 2407 Legal Interface

Pyongyang 2407
Hacking North Korea

```
package gov.no.media.natsign;

public class MnsNative
{
  private static final boolean DBG = true;
  public static final String TAG = "MnsNative";
  static
  {
    System.loadLibrary("medianatsign");
  }
  public static native void getIMEIandIMSI(String paramString1, String paramString2);
  public static native int getNatSignInfoLen(String paramString, int[] paramArrayOfInt);
  public static native int isMagicCorrect(String paramString, int[] paramArrayOfInt);
  public static native int isNatSignFile(String paramString, int[] paramArrayOfInt);
  public static native void saveKeyToFile(byte[] paramArrayOfByte, int paramInt);
  public static native void savePatternToFile(byte[] paramArrayOfByte, int paramInt);
  public static native void saveSelfKeyToFile(byte[] paramArrayOfByte, int paramInt);
}
```

gov.no.media.natsign API

```
package gov.no.media.selfsign;

public class MssNative
{
  private static final boolean D = true;
  public static final String TAG = "MssNative";
  static
  {
    System.loadLibrary("mediaselfsign");
  }
  public static native int CheckSelfSignFile(String paramString, int[] paramArrayOfInt);
  public static native void SetMMSInfoToFile(String paramString, int[] paramArrayOfInt);
  public static native int getSelfSignInfoLen(String paramString, int[] paramArrayOfInt);
  public static native int isSelfResignedByPhone(String paramString, int[] paramArrayOfInt);
  public static native int isSelfSignFile(String paramString, int[] paramArrayOfInt);
  public static native int isSendedByMMS(String paramString, int[] paramArrayOfInt);
  public static native int rsa2048_Sign(byte[] paramArrayOfByte1, byte[] paramArrayOfByte2,
int paramInt, byte[] paramArrayOfByte3);
  public static native int rsa2048_Verify(byte[] paramArrayOfByte1, byte[] paramArrayOfByte2
, byte[] paramArrayOfByte3, int[] paramArrayOfInt);
  public static native int saveSelfSignFile(String paramString, int paramInt);
}
```

gov.no.media.selfsign API

/system/app/RedFlag.apk

Every time a file is accessed, it is checked using Android native code for a signature to see if legal content.
Self-signed files use RSA2048 to store a signature when files are created on the device, e.g. Camera photo.

The following files are used to to create SELFSIGN and to validate NATISIGN files.

- **/data/local/tmp/0.dat** - key file material
- **/data/local/tmp/selfsignkey.dat** - RSA2048 (self-signing key material)
- **/data/local/tmp/legalref.dat** - (IMEI & IMSI of the subscriber, created by saveImeiImsiToFile();
- **/data/local/tmp/pattern.dat** - (files and patterns to match, try to identify .apk or patterns)

kuts.ktr.RFService when “digital signature” checks fail, and a file is not self-signed or state approved, alert.

Every file / folder / removable media access goes through the signature / file checking system.
Any signature alerts are stored in “legals.db”. It stores record of self-signing file history such as photos taken.
It captures details of signature infringements such as attempts to load movies.

전자서명조작체계 (Digital signature manipulation system) 2.0 (RedFlag)

./com/kptc/nv/CertKeyRW.smali
./kuts/ktr/RFService/RFReceiver.smali
./kuts/ktr/RFService/CopyUtils.smali
./kuts/ktr/RFService/RFService.smali
./kuts/ktr/RFService/LegalProvider.smali
./kuts/ktr/RFService/RFScanThread.smali
./kuts/ktr/RFService/BuildConfig.smali
./kuts/ktr/RFService/R.smali
./kuts/ktr/RFService/RFScreenThread.smali
./kuts/ktr/RFService/LegalContract.smali
./gov/no/media/natsign/MnsNative.smali
./gov/no/media/Sign.smali
./gov/no/media/selfsign/MssNative.smali

content://kuts.jsc.providers.legal

전자서명조작체계 native Android components using cryptographic functions (RSA2048) to watermark all created documents and check/scan files for signatures.

Android content provider to SQLite (legals.db). Traceviewer can read/export contents.

Broadcast Reciever MEDIA_MOUNTED, BOOTUP & SHUTDOWN for 전자서명조작체계 RFService

Android kuts.ktr.RFService

Pyongyang 2407
Hacking North Korea

Debugging enabled in some 전자서명조작체계 components

```
public static String[] Video_extensions = { "3gp", "amv", "avi", "avc", "f4v",
"flv", "mov", "mp4", "mpeg", "mpg", "mpeg4", "m1v", "wmv", "wma", "m2v", "mpv2",
"rm", "rmvb", "vob" };

public static String[] extensions = { "3g2", "3gp2", "3gpp", "aac", "ac3", "amr",
"ape", "asf", "awb", "asx", "bmp", "bik", "cda", "csf", "cda", "cue", "dat",
"divx", "doc", "docx", "dts", "evo", "flac", "gif", "hlv", "htm", "html", "ifo",
"ivm", "jpeg", "jpg", "kpl", "m1a", "m2a", "m2p", "m2t", "m2ts", "m3u", "m4a",
"m4b", "m4p", "m4r", "m4v", "mid", "midi", "mka", "mkv", "mmf", "mp2", "mp2v",
"mp3", "mpa", "mpc", "mpe", "mod", "mts", "ofr", "ogg", "ogm", "pcx", "pdf",
"png", "ppt", "pmp", "pptx", "pss", "pva", "pls", "qpl", "qt", "ra", "ram", "rp",
"rpm", "rt", "rtf", "smf", "swf", "smi", "smil", "scm", "smpl", "tga", "tif",
"tpr", "tiff", "tp", "ts", "tta", "txt", "tod", "vp6", "wav", "wv", "wm", "wmp",
"webm", "xls", "xlsx", "xml" };
```

RFScan Signature'd File Exts.

Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: browser_history

	_id	path	size	date_modified	isvalid
	Filter	Filter	Filter	Filter	Filter
1	1	/storage/sdcard0/DCIM/Camera/IMG_20140106_200922.jpg	2774753	1389038275104	1
2	2	/storage/sdcard0/DCIM/Camera/IMG_20140106_201131.jpg	3203553	1389038275321	1
3	3	/storage/sdcard0/DCIM/Camera/IMG_20140107_021105.jpg	2965729	1389038275406	1
4	4	/storage/sdcard0/DCIM/Camera/VID_20140107_031533.3gp	35604822	1389038275571	1
5	5	/storage/sdcard0/DCIM/Camera/VID_20140107_031813.3gp	44422147	1389038275702	1
6	6	/storage/sdcard0/DCIM/Camera/VID_20140107_034709.3gp	1205374	1389038275906	1
7	7	/storage/sdcard0/DCIM/Camera/IMG_20140107_034723.jpg	3439201	1389038276005	1
8	8	/storage/sdcard0/DCIM/Camera/IMG_20140107_034745.jpg	1945313	1389038276173	1
9	9	/storage/sdcard0/DCIM/Camera/IMG_20140107_034750.jpg	2046177	1389038276298	1
10	10	/storage/sdcard0/DCIM/Camera/IMG_20140115_041924.jpg	1008941	1390152767996	1
11	11	/storage/sdcard0/DCIM/Camera/IMG_20140115_041925.jpg	1033645	1390152768110	1
12	12	/storage/sdcard0/DCIM/Camera/IMG_20140115_041928.jpg	1001901	1390152768202	1

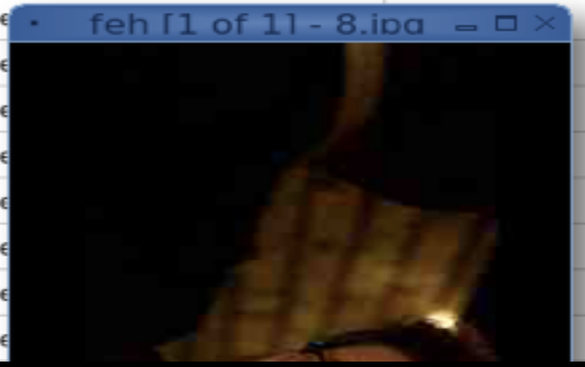
TraceViewer (browser_history)

Pyongyang 2407
Hacking North Korea

```
144         bzero(&(sa.sin_zero),8);
145       printf("[ connecting to %s %d/tcp\n",server,port);
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161       host = gethostbyname(server);
162
163
164
165         exit(0);
166       }
```

Table: capture_list

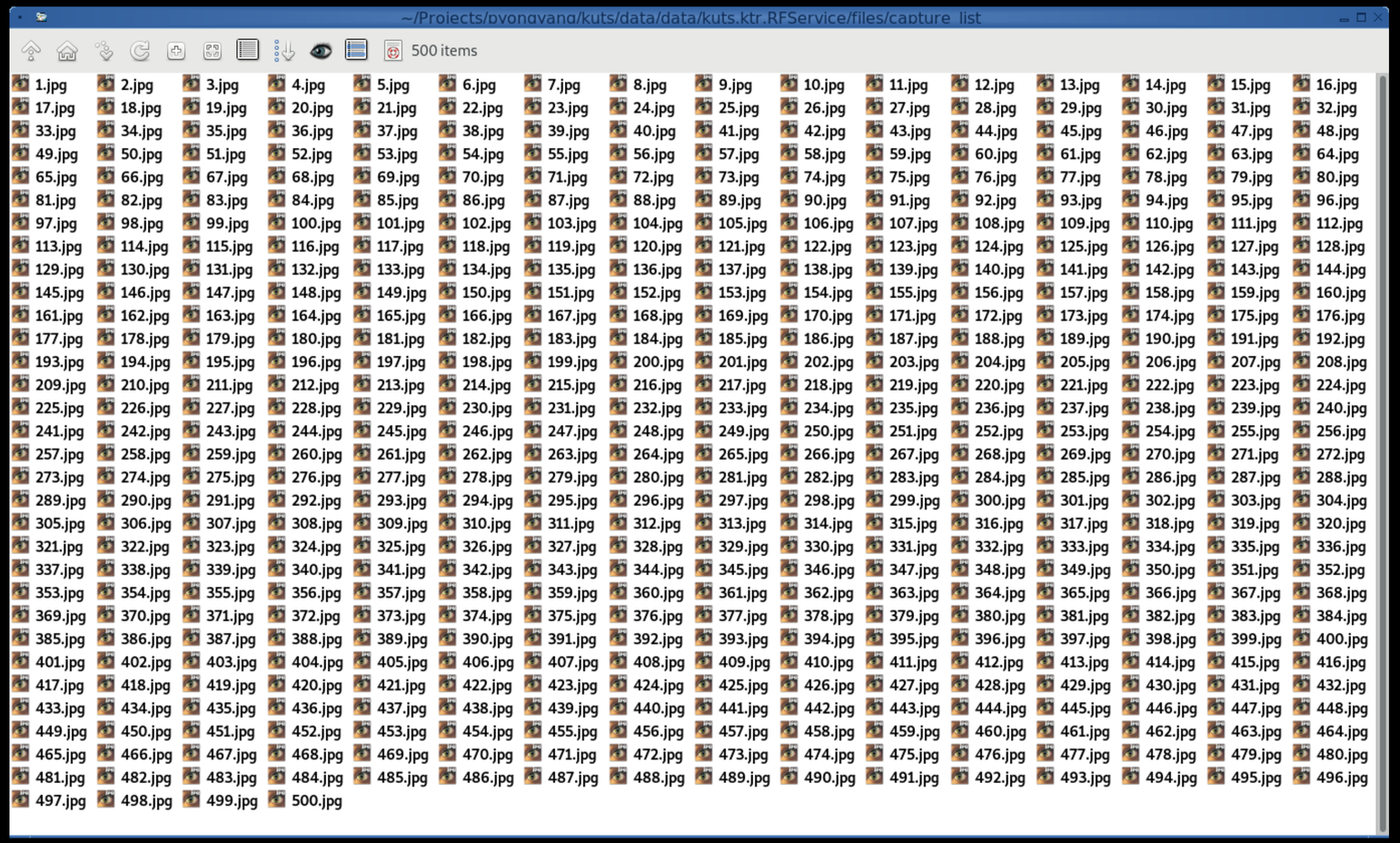
	_id	apk_name	captured_time	_data	isvalid
	Filter	Filter	Filter	Filter	Filter
1	1	KPubReader	138879847...	/data/data/kuts.ktr.RFService/files/capture_list/1.jpg	1
2	2	drozer Agent	139031109...	/data/data/kuts.ktr.RFService/files/capture_list/2.jpg	1
3	3	MX Player	139033393...	/data/data/kuts.ktr.RFService/files/capture_list/3.jpg	1
4	4	MX Player	138853880...	/data/data/kuts.ktr.RFService/files/capture_list/4.jpg	1
5	5	MX Player	138854384...	/data/data/kuts.ktr.RFService/files/capture_list/5.jpg	1
6	6	MX Player	138871374...	/data/data/kuts.ktr.RFService/files/capture_list/6.jpg	1
7	7	Video	138871674...	/data/data/kuts.ktr.RFService/files/capture_list/7.jpg	1
8	8	Video	138871704...	/data/data/kuts.ktr.RFService/files/capture_list/8.jpg	1
9	9	Video	138871734...	/data/data/kuts.ktr.RFService/files/capture_list/9.jpg	1
10	10	Video	13887	files/capture	
11	11	Video	13887	files/capture	
12	12	Video	13887	files/capture	
13	13	Video	13887	files/capture	
14	14	Video	13887	files/capture	
15	15	Video	13887	files/capture	
16	16	Video	13887	files/capture	
17	17	Video	13887	files/capture	



TraceViewer (capture_list)

열람리력

Pyongyang 2407
Hacking North Korea



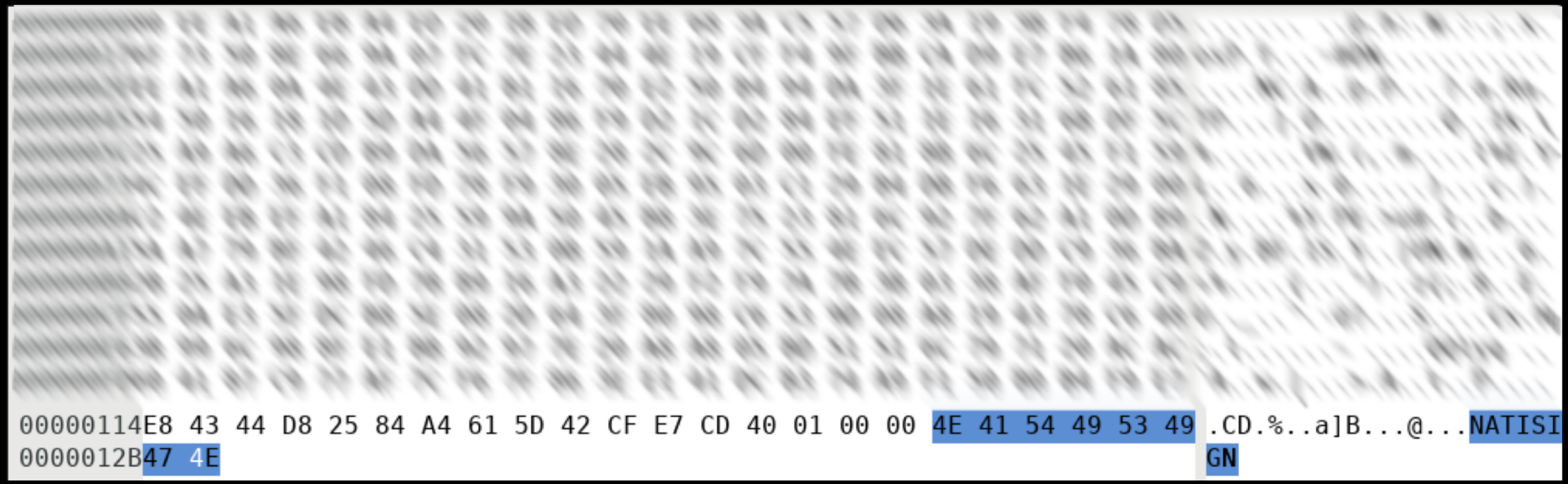
Files not self-signed or signed by the state, when accessed or viewed, are recorded in a database.

Screen shots are taken of the offending access.

RedFlag History (capture_list)

Pyongyang 2407
Hacking North Korea

/프로그람자료/.KwangMyong/도움말.pdf



Example of NATISIGN file

/DCIM/Camera/IMG_2014_0103_085922.jpg

Watermarking contains IMEI/IMSI in the process. SELFSIGN uses a RSA2048 key process along with legalref.dat. Its appended to any file created on the device.

D/gov_sign(2070): --------.jpg------4----
D/gov_sign(2070): get_flength : file name = /storage/sdcard0/DCIM/Camera/
IMG_20140103_085922.jpg, fSize = 820065, POSTFIX_LEN = 0
D/gov_sign(2070): MnsNative isNatSignFile : file name = /storage/sdcard0/DCIM/Camera/
IMG_20140103_085922.jpg, ***result = 264***
E/MediaSelfSign(2070): Call isSelfSignFile()---------- successed!
E/MediaSelfSign(2070): MSSMediaVerifyFileByPath()---------! return 0
E/MediaSelfSign(2070): MSSMediaVerifyFileByPath()---------! return 0
E/MediaSelfSign(2070): isSelfSignFile()---------successed! Err = 0, return 1

Example of SELFSIGN file

```
; =============== B E G I N N I N G   O F   P R O C E D U R E ===============

; Variables:
;    var_18: int32_t, -24


        isNatSignFile:
00001978        push        {r4, r5, lr}
0000197a        ldr         r1, [r0]
0000197c        movs        r3, #0xa9
0000197e        lsls        r3, r3, #0x2
00001980        sub         sp, #0xc
00001982        ldr         r3, [r1, r3]
00001984        adds        r1, r2, #0x0
00001986        movs        r2, #0x0
00001988        blx         r3
0000198a        adds        r5, r0, #0x0
0000198c        bl          verifyFileSign                          ; verifyFileSign
00001990        ldr         r1, =0x3e52                             ; dword_19ac,0x3e52
00001992        ldr         r2, =0x3e9c                             ; dword_19b0,0x3e9c
00001994        adds        r4, r0, #0x0
00001996        str         r0, [sp, #0x18 + var_18]
00001998        adds        r3, r5, #0x0
0000199a        add         r1, pc                                  ; "gov_sign"
0000199c        add         r2, pc                                  ; "MnsNative isNatSignFile : file name = %s, result = %d"
0000199e        movs        r0, #0x3
000019a0        blx         __android_log_print@PLT                 ; __android_log_print
000019a4        add         sp, #0xc
000019a6        adds        r0, r4, #0x0
000019a8        pop         {r4, r5, pc}
              ; endp
```

평양 2407 libmedianatsign.so

Pyongyang 2407
Hacking North Korea

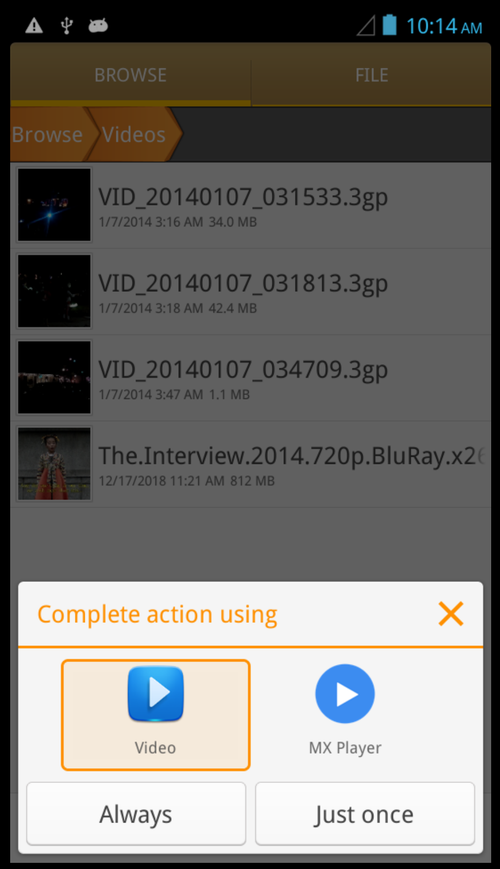
```
144         bzero(&(sa.sin_zero),8);
145     printf("[ connecting to %s %d/tcp\n",server,port);
146
```

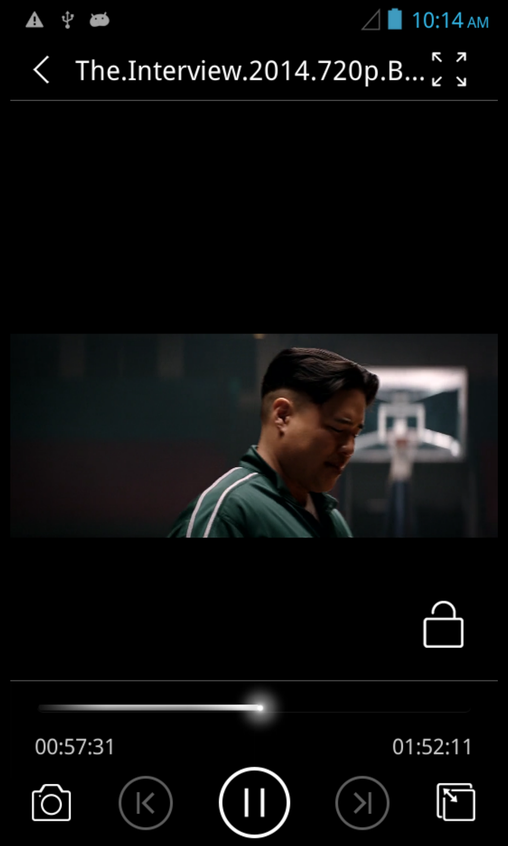
```
            isNatSignFile:
00001978        push        {r4, r5, lr}
0000197a        ldr         r1, [r0]
0000197c        movs        r3, #0xa9
0000197e        lsls        r3, r3, #0x2
00001980        sub         sp, #0xc
00001982        ldr         r3, [r1, r3]
00001984        adds        r1, r2, #0x0
00001986        movs        r2, #0x0
00001988        blx         r3
0000198a        adds        r5, r0, #0x0
0000198c        movs        r0, #0x1
0000198e        nop
00001990        ldr         r1, =0x3e52                 ; dword_19ac,0x3e52
00001992        ldr         r2, =0x3e9c                 ; dword_19b0,0x3e9c
00001994        movs        r4, #0x1
00001996        str         r0, [sp]
00001998        adds        r3, r5, #0x0
0000199a        add         r1, pc                      ; "gov_sign"
0000199c        add         r2, pc                      ; "MnsNative isNatSignFile : file name = %s, result = %d"
0000199e        movs        r0, #0x3
000019a0        blx         __android_log_print@PLT     ; __android_log_print
000019a4        add         sp, #0xc
000019a6        movs        r0, #0x1
000019a8        pop         {r4, r5, pc}
```

```
161     host = gethostbyname(server);
162
163
164
165         exit(0);
166     }
```

평양 2407 isNatSignFile patched

Pyongyang 2407
Hacking North Korea





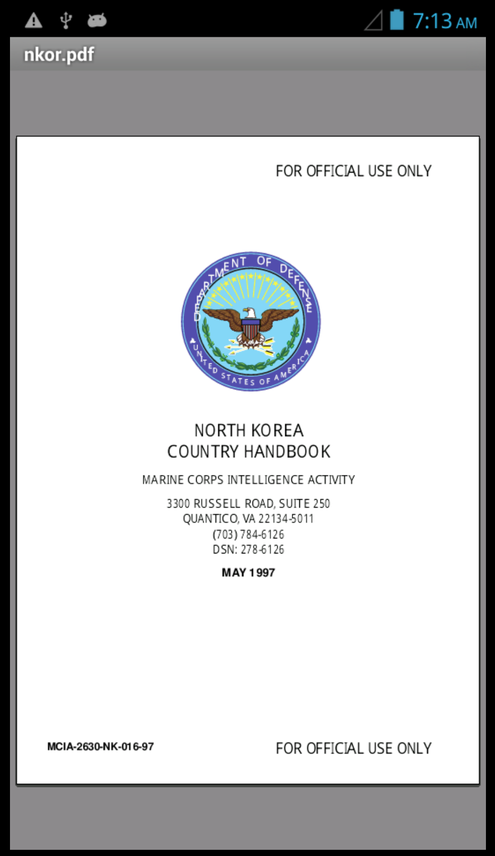
Exploited gov_sign logic, DPRK government approved media

Phone now treats all media as Nat_Sign_File bypassing 전자서명조작체계 validation

Can load any PDF, MP4, JPG etc - phone treats it as DPRK approved state media, no longer logs or alerts on use.

전자서명조작체계 gov_sign exploit

Pyongyang 2407
Hacking North Korea



I watched The.Interview.2014 on 평양 2407 device.

I loaded documents of interesting facts about North Korea onto the device.

“adb logcat” shows all files are treated as Nat_Sign_File.

Patch files & jailbreak tool - https://github.com/hackerhouse-opensource/pyongyang_2407

전자서명조작체계 Nat_Sign_File.

MT6582 & MT6627

com.mediatek.bluetooth

MT6627 is the mediatek companion RF chipset.

Bluetooth, GPS, FM-Radio & 802.11 **ONLY**

https://android.googlesource.com/kernel/mediatek/+refs

drivers/misc/mediatek/conn_soc/common/core/wmt_*

- **Bluetooth**
  - Bluetooth specification v2.1+EDR
  - Bluetooth specification 3.0+HS compliance
  - Bluetooth v4.0 Low Energy (LE)
  - Integrated PA with 10dBm (class 1) transmit power and Balun
  - Rx sensitivity: GFSK -95dBm, DQPSK -94dBm, 8-DPSK -88dBm
  - Best-in-class BT/Wi-Fi coexistence performance
  - Up to 4 piconets simultaneously with background inquiry/page scan
  - Supports Scatternet
  - Packet loss concealment (PLC) function for better voice quality
  - Low-power scan function to reduce the power consumption in scan modes

MEDIATEK MT6627 RF CHIP

```
[bluetooth]# info E6:68:46:EE:41:F2
Device E6:68:46:EE:41:F2
        Name: 평양 2407
        Alias: 평양 2407
        Class: 0x5a020c
        Icon: phone
        Paired: yes
        Trusted: yes
        Blocked: no
        Connected: no
        LegacyPairing: no
        UUID: Dialup Networking         (00001103-0000-1000-8000-00805f9b34fb)
        UUID: OBEX Object Push          (00001105-0000-1000-8000-00805f9b34fb)
        UUID: Audio Source              (0000110a-0000-1000-8000-00805f9b34fb)
        UUID: A/V Remote Control Target (0000110c-0000-1000-8000-00805f9b34fb)
        UUID: Advanced Audio Distribu.. (0000110d-0000-1000-8000-00805f9b34fb)
        UUID: Headset AG                (00001112-0000-1000-8000-00805f9b34fb)
        UUID: NAP                       (00001116-0000-1000-8000-00805f9b34fb)
        UUID: GN                        (00001117-0000-1000-8000-00805f9b34fb)
        UUID: Handsfree Audio Gateway   (0000111f-0000-1000-8000-00805f9b34fb)
        UUID: Phonebook Access Server   (0000112f-0000-1000-8000-00805f9b34fb)
        UUID: PnP Information           (00001200-0000-1000-8000-00805f9b34fb)
        Modalias: bluetooth:v0046p0802d0903
```

```
[NEW] Device E6:68:46:EE:41:F2 평양 2407
[bluetooth]# devices
Device E6:68:46:EE:41:F2 평양 2407
[bluetooth]# connect E6:68:46:EE:41:F2
Attempting to connect to E6:68:46:EE:41:F2
[CHG] Device E6:68:46:EE:41:F2 Connected: yes
[평양 2407]# connect E6:68:46:EE:41:F2
Attempting to connect to E6:68:46:EE:41:F2
Connection successful
[평양 2407]# devices
[CHG] Device E6:68:46:EE:41:F2 Connected: no
```

평양 2407 Bluetooth Profile

Pyongyang 2407
Hacking North Korea

Bluetooth Attacks

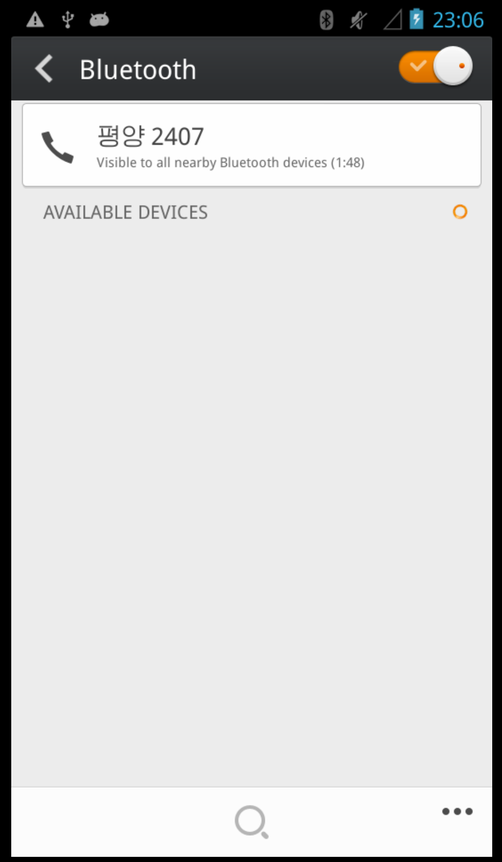
평양 2407 file transfer

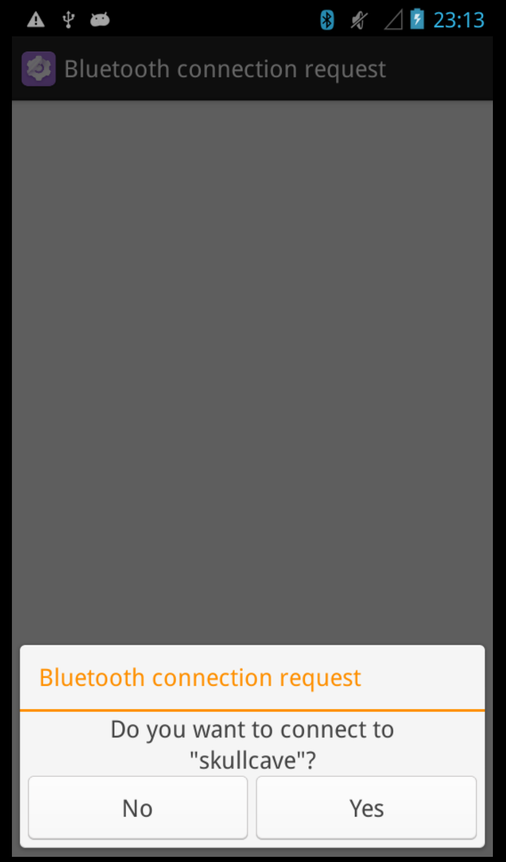
CVE-2017-0781 - CVE-2017-785 - Android Blueborne(?)

MediaTek wrote own Bluetooth HAL/interface for MT6627.

Once a bluetooth device is paired, connections are always accepted from that device. Entering “no” to “Bluetooth connection request” still allows connection regardless until user forcefully selects “Unpair”

Default Bluetooth name is “평양 2407” discoverable for 2 minutes on activation and user forced scanning only.





The following North Korean providers are known to have been active.

- Koryolink (고려링크) 467/05
- Kang Song NET 467/06
- Byol (별) Star

koryolink™

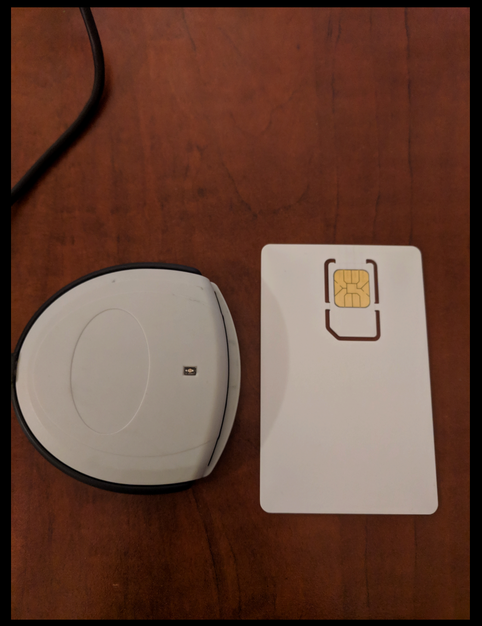
Koryolink SIM cards permitted to buy for foreigners. +850 0 191 xxxx

North Korea - KP [edit]

MCC	MNC	Brand	Operator	Status	Bands (MHz)	References and notes
467	05	Koryolink	Cheo Technology Jv Company	Operational	UMTS 2100	for foreigners
467	06	Koryolink	Cheo Technology Jv Company	Operational	UMTS 2100	for DPRK citizens
467	193	SunNet	Korea Posts and Telecommunications Corporation	Not operational	GSM 900	

Network Infrastructure (MCC)

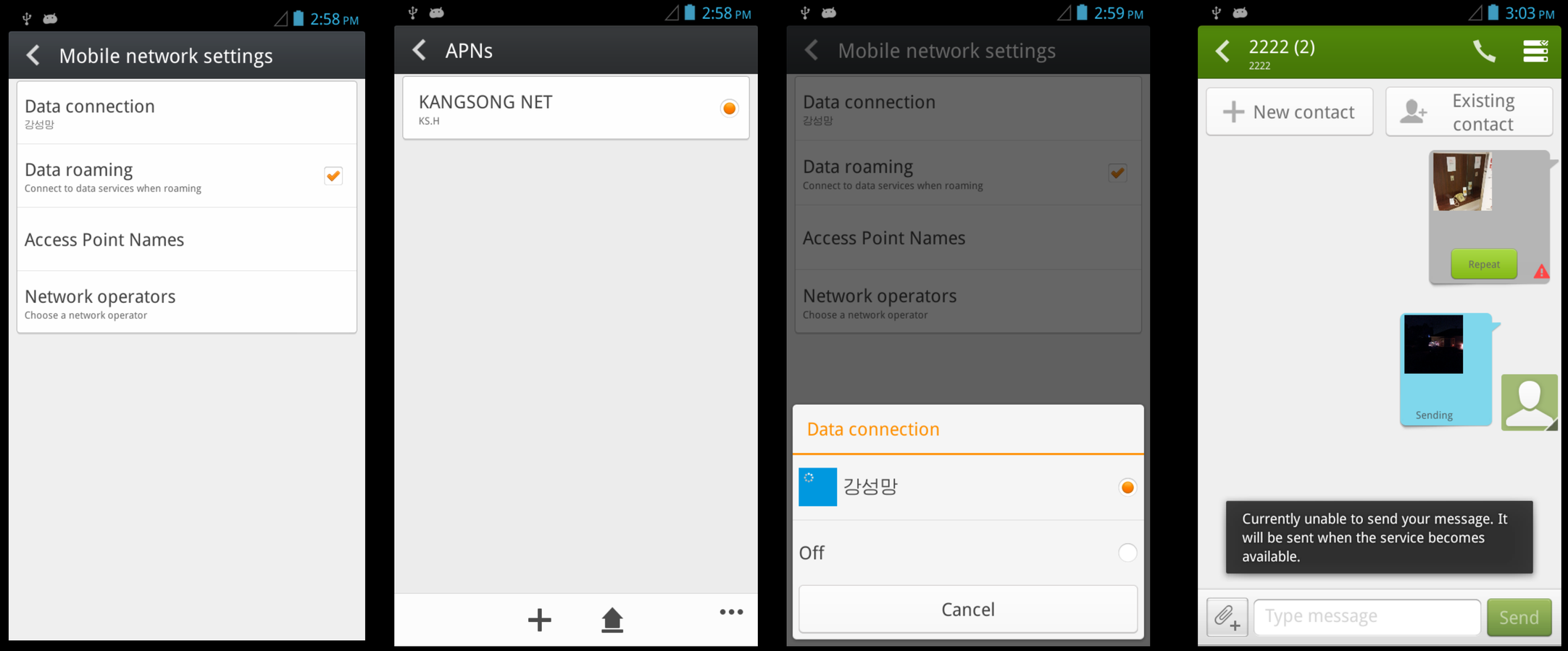
```
./pySim-prog.py -p1 -t grcardsim -n Koryolink -i 467060000000000 -s 00000000000000000000 -x 467 -y 06
Insert card now (or CTRL-C to cancel)
Generated card parameters :
 > Name     : Koryolink
 > SMSP     : e1ffffffffffffffffffffffff0581005155f5ffffffffffff000000
 > ICCID    : 00000000000000000000
 > MCC/MNC  : 467/6
 > IMSI     : 467060000000000
 > Ki       : 12db6bfdd5cb8c8af844a3b6b65a9f49
 > OPC      : ef11dc6f2fdc9aed6bd16c2fd28d4f3d
 > ACC      : None

Programming ...
Done !
```



Phone can send MMS…. no SIM error.
Clone Kang Song NET SIM to test with…
Works to test MMS and other network functions…
Kang Song NET is for **DPRK citizens only**..

Over-The-Air Network Attacks

Pyongyang 2407
Hacking North Korea



Kang Song Net APN details

Pyongyang 2407
Hacking North Korea

- **MediaTek bugdoors** - features for hacking/access
- **Red Flag** Digital Watermarking & Signature System
- Limited **USB/Bluetooth/MMS**/WiFi connectivity
- Digital media is state controlled inside **North Korea**
- **Exploits** required for unauthorised digital media use
- Mobile technology is supplied by **China**
- No platform diversity, attack techniques are portable

Summary



Pyongyang 2407
Hacking North Korea

Questions & Thanks!

Pyongyang 2407 ROM, tools, documents & exploits

https://github.com/hackerhouse-opensource/pyongyang_2407

https://github.com/hackerhouse-opensource/pyongyang_2407#acknowledgements

https://hacker.house