
硬件黑客

Who am I

刘凯仁

- 刘凯仁
- 1981
- 国际 马来西亚
- 籍贯 潮汕
- 创始人

PANDA SECURITY

西班牙电脑防毒公司

亚太区技术总监

- 亚太区病毒实验室
- 亚太区病毒事件
- 亚太区重要客户管理

自由黑客

沉迷于各种软硬件逆向

- 逆向商业软件
- 逆向物联网
- 逆向网络安全

HITB

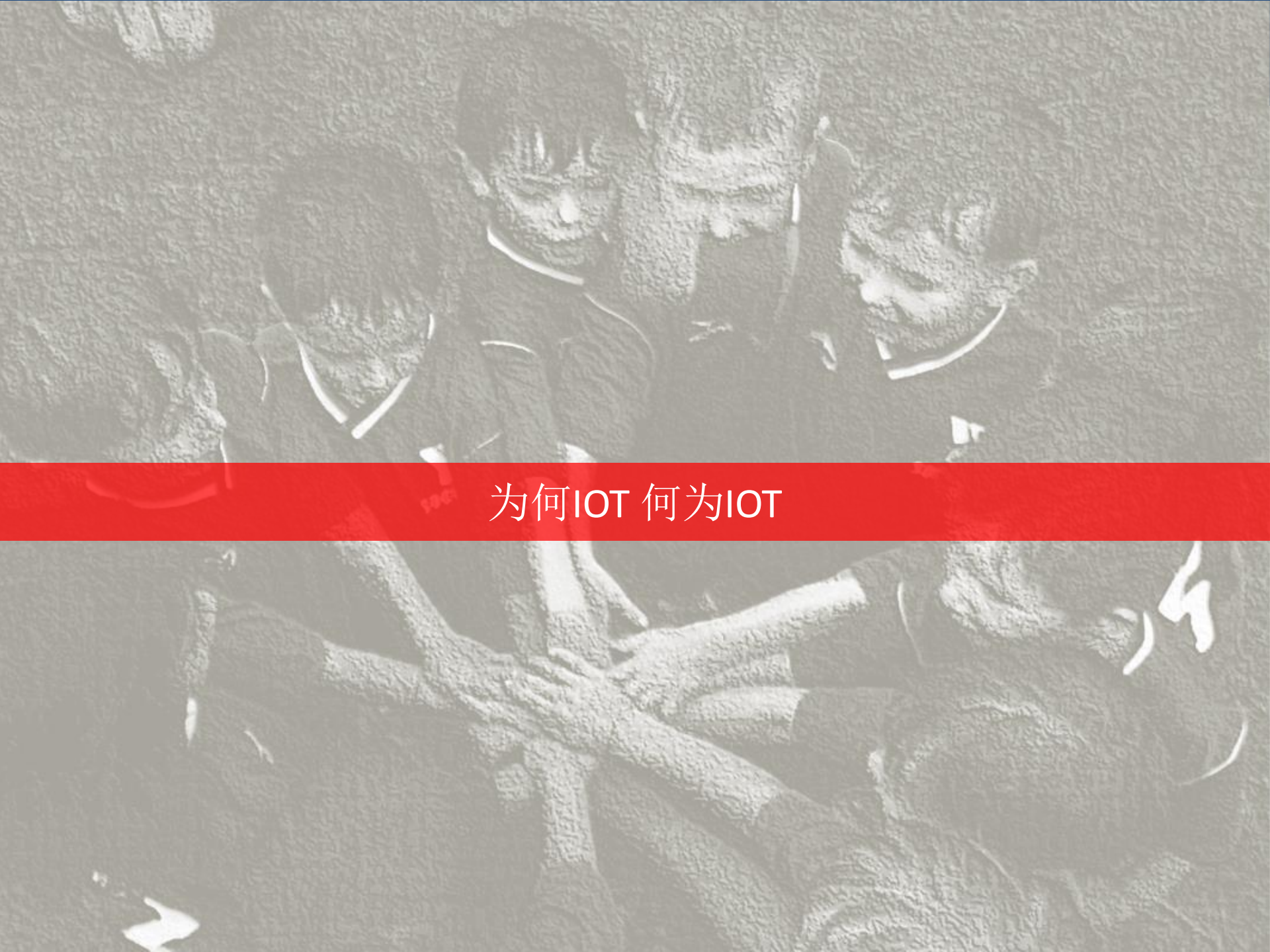
年度国际黑客峰会

Hack in the box 主要在荷兰和新加坡举行

- 2006 – 现在
- 主要成员
- 黑客攻防大赛主要成员

- 2005, HITB CTF, 马来西亚黑客大赛, 冠军名 共20多个国际队伍
- 2010, Hack In The Box 马来西亚, 演讲嘉宾
- 2012, Codegate 韩国,演讲嘉宾
- 2015, VXRL 香港,演讲嘉宾
- 2015, HITCON 台湾黑客大赛预赛, 前10名 共4千多个国际队伍
- 2016, Codegate 韩国黑客大赛预赛, 前五名 共3千多个国际队伍
- 2016, Qcon 北京,演讲嘉宾
- 2016, Kcon 北京, 演讲嘉宾
- 2016, 国际反病毒大会 天津, 演讲嘉宾

- 苹果系统漏洞提升
- GDB 软件漏洞
- Metasploit 模块开发
- Linux 系统安全逃逸
- http://www.githiub.com/xwings/tuya
- 微博: @kaijern


为何IOT 何为IOT

IOT是生活

网络物品

- 摄像头
- 空调
- 电视机
- 排插
- 风扇
- 热水器
- 电饭煲
- 冰箱
- 手表
- 门锁
- 防盗
- 手机

安全科技公司

黑客如何攻击物联网

全球2.5万网络摄像机被黑，用于构建DDOS攻击僵尸网络

dawner 2016-06-30 +5 共152860人围观，发现 9 个不明物体 资讯

最近的一份研究报告显示，黑客攻破了大量的网络摄像机，在网上发起DDoS攻击，有目的地对那些大型网站进行打击。被黑的网络摄像机IP地址分布在世界各地，不少于105个国家，总数量超过2.5万。

由于安全标准和管控制度的缺失，目前脆弱的网络嵌入式设备，包括智能电视、冰箱、微波炉、机顶盒，摄像机以及联网打印机经常会被黑客盯上。我们曾看到过黑客攻破超过10万的智能电视和智能冰箱，然后借助它们发送了数以百万计的垃圾邮件。同时，我们也曾见过黑客利用黑掉的打印机和机顶盒去开采比特币。

网络摄像机分析

安全公司Sucuri的研究员们在帮助一个小型珠宝店网站抵御针对性的DDoS攻击的时候，遇到了超过2.5万网络摄像机组成的僵尸网络。

当时，该珠宝店网站收到了3.5万次/秒的HTTP请求，以至于正常的用户完全无法对网站进行访问。然而，当技术人员试图通过某网络寻址和路由系统（Anycast任播）去防护洪水攻击时，该僵尸网络骤然增加了HTTP请求的频率，攻击飙升到了5万次/秒。

安全研究人员表示，这次来自HTTP层面的DDoS攻击让WEB服务器不堪重负，服务器资源耗尽后，网站自然就崩溃了。这次DDoS攻击持续了好几天，让研究人员对它们的来源有了不少兴趣。后来，研究人员发现这些来自远程网络摄像机的请求包，已经被黑客拿来攻击其他服务了。

Sucuri的CTO Daniel Cid表示：

> “其实，这并不是黑客首次使用物联网设备去进行DDoS攻击了。但是我们还没遇到过单纯使用网络摄像机设备进行的攻击，而且请求还如此持久的情况。”

这些来自网络摄像机IP地址分布在世界各地，不少于105个国家。Sucuri研究人员指出，在当时的几个小时之内，至少出现了25513个独立IP，其中有些地址是IPV6。

总结分析

这并不是黑客第一次劫持网络摄像机发的DDoS攻击。在去年年底，Imperva的Incapsula团队曾示警，有一大波运行了嵌入式Linux和BusyBox工具包的网络摄像机，组成了庞大的僵尸网络。由于物联网的迅速增长，我们使用技术的方法也有了改变。它大大拓宽了黑客的攻击面，从安全的角度来讲，物联网的确能让人心生不安。

*参考来源：TA，FB小编dawner编译，转载请注明来自FreeBuf黑客与极客（FreeBuf.COM）

酒店三次遭遇黑客勒索，开房顾客被拒之门外

本文作者：谢幺 2017-02-01 19:28

导语：酒店行业遭黑客攻击在国外早已屡见不鲜，望国内相关从业者引以为戒。

人常说：“别在一个坑里跌两次”，但奥地利的“Jägerwirt 浪漫酒店”不幸被黑客以相同的伎俩勒索了三次。

【被勒索的 Jägerwirt 酒店】

在如今的酒店行业里，现代化的 IT 系统和联网的电子门禁系统十分常见。酒店极尽所能地利用现代化设备来优化住房体验，但他们常常忽视黑客的存在。

勒索者的三进三出

据被攻击的酒店经理表示，第一次遭遇黑客攻击是在几个月前，黑客渗透进了他们的电子门禁系统，将数百位客人锁在门外，要求酒店方支付 1600 美元（约11000人民币）赎金才能解锁房门。当时他为了尽快恢复酒店的正常运营秩序，便直接向对方支付了赎金。

想象一下，如果你是那个酒店经理，一群进不了房门的顾客正心急如焚地等你为他们开门，而勒索者还打算不断抬高勒索金。这时你有两种做法：一是报警，然后等警察花几天时间慢慢和黑客周旋，这期间酒店直接歇业并赔偿顾客损失，另一个选择是直接支付不算太高的赎金，相信大多数人都会选择后者。

然而酒店经理万万没想到，虽然黑客收到赎金后解锁了酒店房门，但在酒店的 IT 系统中留了一个安全后门。不久之后，黑客卷土重来，用相同的方法进行了第二次勒索，再一次导致180多位顾客无法进入房间。

面对第二次勒索，酒店经理虽然报了警，但为了酒店正常运营，他们依然选择支付赎金，赶来的警察也对此无能为力。有趣的是，酒店并没有意识到 IT 系统的后门依然存在，直到第三次遭遇勒索。

据雷锋网了解，由于每一次黑客勒索的金额都不是很高，均在1500 美元（约11000元人民币）左右，因此酒店大多选择直接支付赎金，由于根本不知道黑客是谁，酒店也无法从保险公司那里得到任何赔偿。

所幸的是，目前该酒店已经计划替换掉所有出过问题的计算机，并按照最新的网络安全标准重新部署集成，他们将在新的系统中取消掉所有不必要的网络连接，以确保安全。

最关键的是，他们正在计划为每一道房门配一套真正的钥匙锁。

（公众号：雷锋网）

雷锋网原创文章，未经授权禁止转载，详情见转载须知

15万台打印机被黑，打印出了一堆奇怪的东西

bimeover 2017-02-09 共37702人围观，发现 11 个不明物体 资讯

近日，国外有一个自称“stackoverflowin”的黑客侵入了超过15万台打印机。被入侵的这些打印机全部都打印出了这名黑客留下的警告信息。

不过据他本人声称，他控制这些打印机的目的是为了提高人们对打印机安全的认识，打印机在安全这块儿实在太薄弱了。如果你的打印机会不受控制地打印“YOUR PRINTER HAS BEEN PWND'D”那就说明你的打印机已经沦陷了。

这次“攻击”其实只是个脚本

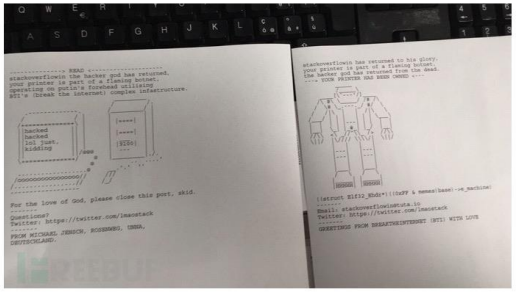
在24小时内，Stackoverflowin就跑着一个他心爱的小脚本，搜索打开的打印机端口，搜索到符合条件的打印机给该打印机发送一个打印作业。大到企业总部的多功能打印机，小到餐馆收据打印机多多少少都受到了影响。

Stackoverflowin会控制打印机输出各种信息，最新的一条是这样写的：

> 你们的黑客之神stackoverflowin回来了，这台打印机已经是火焰僵尸网络的肉鸡之一了，想办法修好你们的打印机吧。有问题？推特找我。

```
stackoverflowin the hacker god has returned, your printer is part of a flaming botnet, operating on putin's forehead utilising BTI's (break the internet) complex infrastructure.

[ASCII ART HERE]

For the love of God, please close this port, skid.

--------

Questions?

Twitter: https://twitter.com/lmaostack

--------
```

第一个版本打印的消息上有一个机器人模样的ASCII码，还列出了这个灰帽子的邮箱地址。这个版本的图案则像是一个机箱旁边配上一台打印机。就像下图这样。

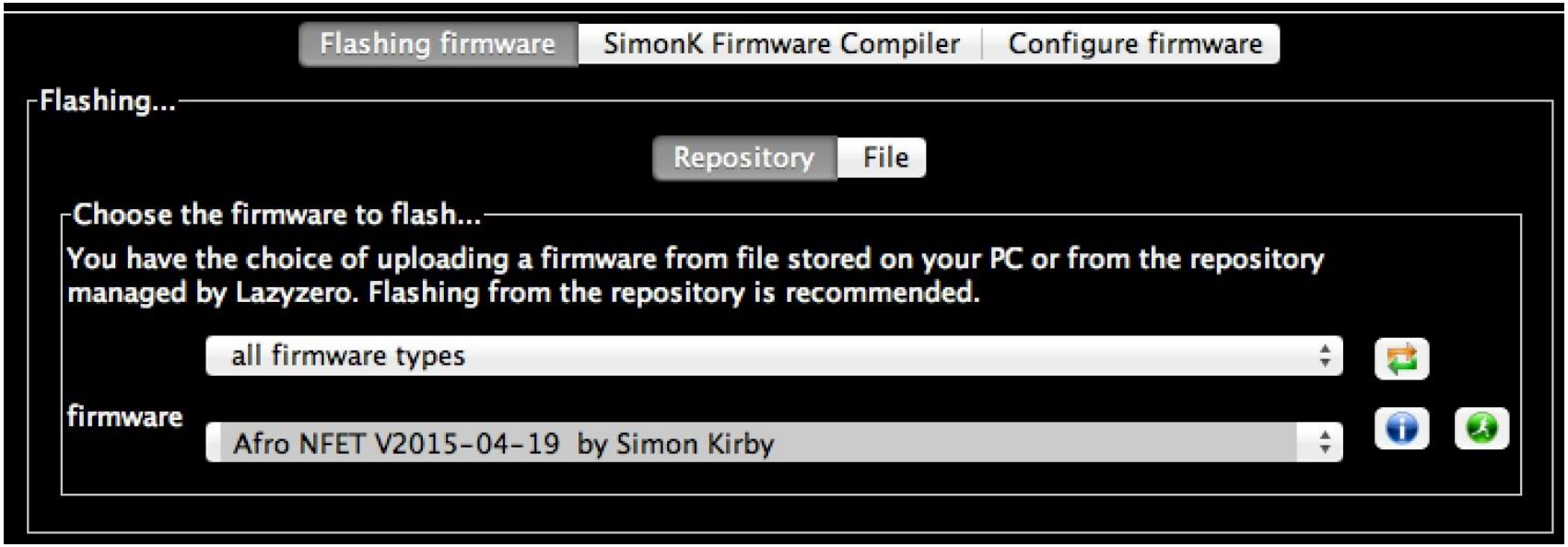



大男孩的玩具

黑客式开箱

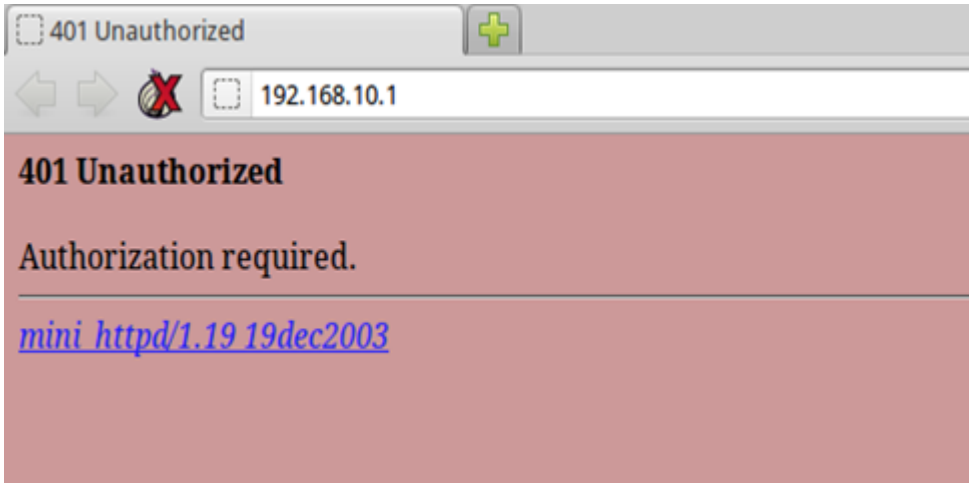

当世界和平时

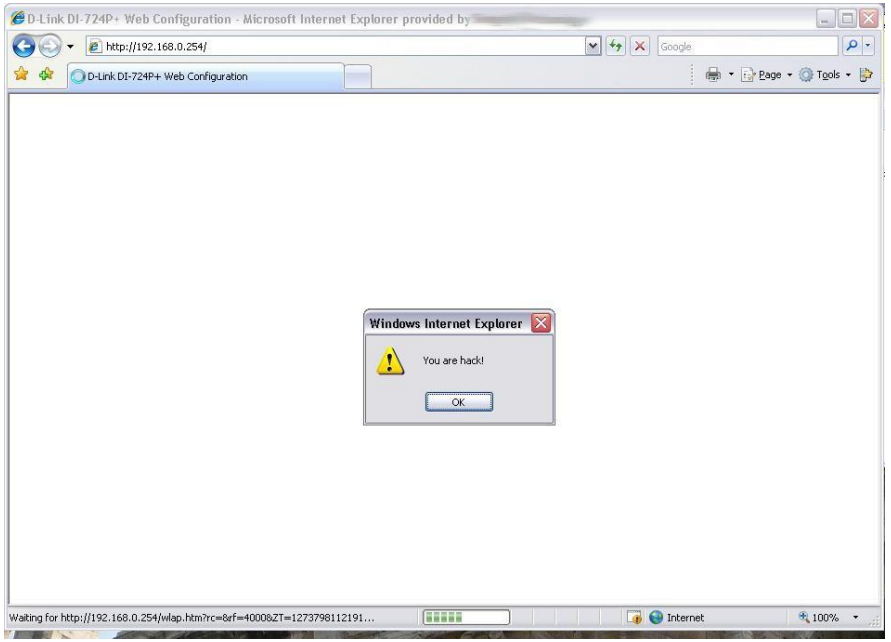
最初的目的


Flashing firmware
SimonK Firmware Compiler
Configure firmware
Flashing...
Repository
File
Choose the firmware to flash...
You have the choice of uploading a firmware from file stored on your PC or from the repository managed by Lazyzero. Flashing from the repository is recommended.
all firmware types
firmware
Afro NFET V2015-04-19 by Simon Kirby

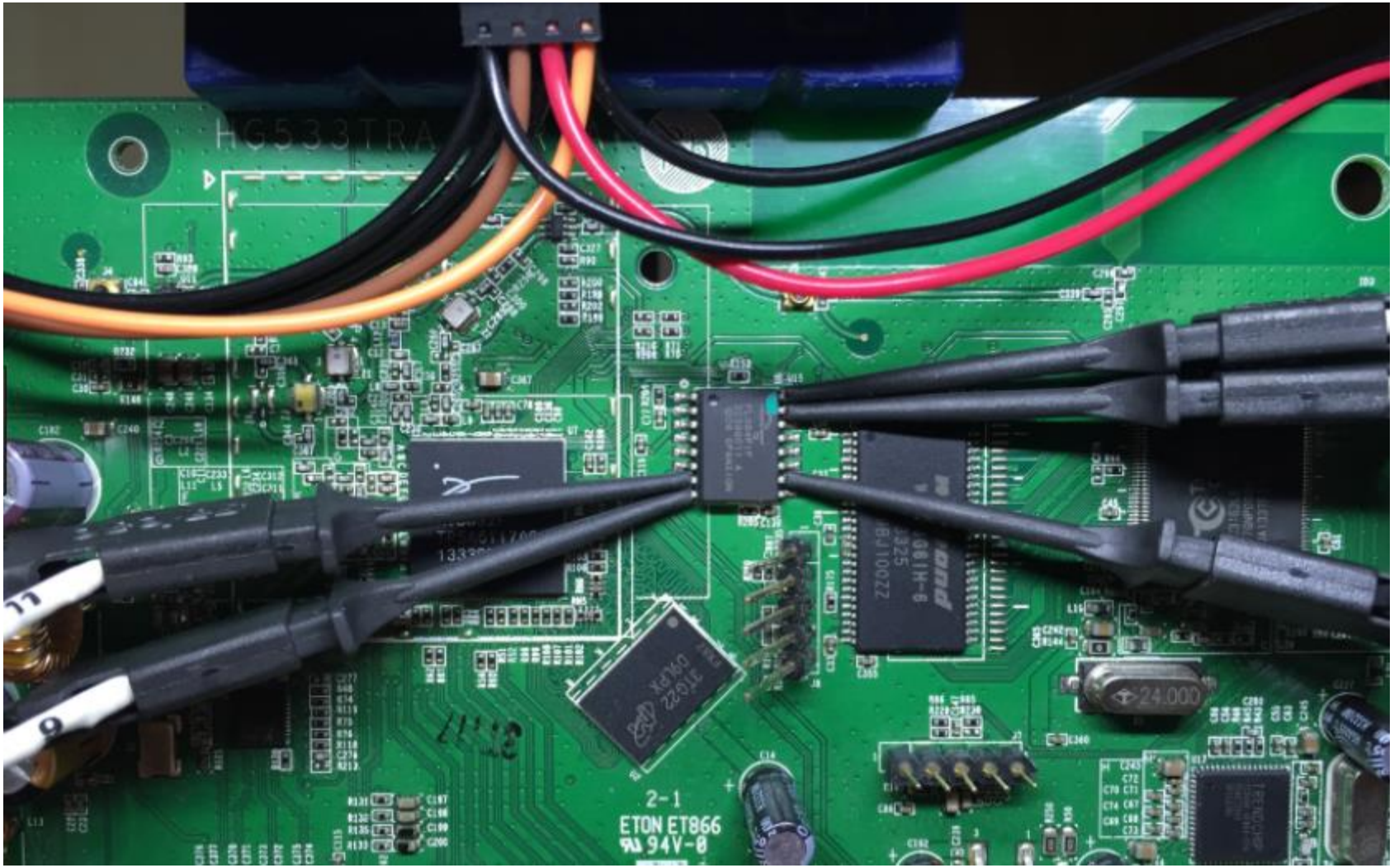

门派之分

软件调试


401 Unauthorized
192.168.10.1
401 Unauthorized
Authorization required.
mini_httpd/1.19 19dec2003

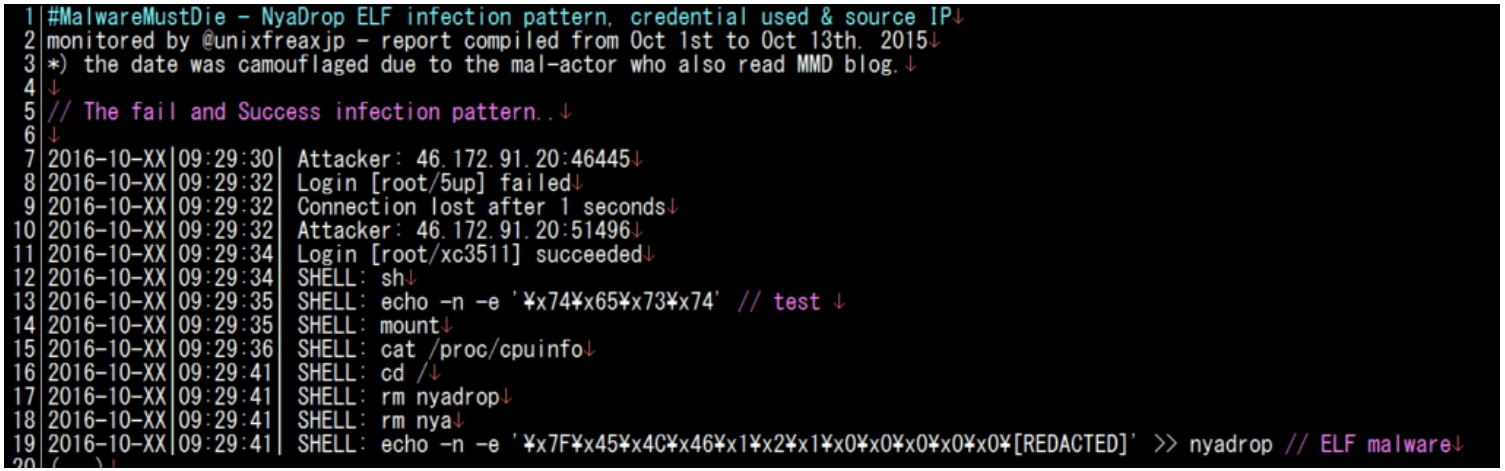

D-Link DI-724P+ Web Configuration - Microsoft Internet Explorer provided by
http://192.168.0.254/
D-Link DI-724P+ Web Configuration
Windows Internet Explorer
You are hack!
OK
Waiting for http://192.168.0.254/wlap.htm?rc=8rf=40008ZT=1273798112191...
Internet
100%

硬件调试



传统软件调试

软件调试


#MalwareMustDie - NyaDrop ELF infection pattern, credential used & source IP
monitored by @unixfreaxjp - report compiled from Oct 1st to Oct 13th, 2015
*) the date was camouflaged due to the mal-actor who also read MMD blog.
// The fail and Success infection pattern..
2016-10-XX|09:29:30| Attacker: 46.172.91.20:46445
2016-10-XX|09:29:32| Login [root/5up] failed
2016-10-XX|09:29:32| Connection lost after 1 seconds
2016-10-XX|09:29:32| Attacker: 46.172.91.20:51496
2016-10-XX|09:29:34| Login [root/xc351l] succeeded
2016-10-XX|09:29:34| SHELL: sh
2016-10-XX|09:29:35| SHELL: echo -n -e '¥x74¥x65¥x73¥x74' // test
2016-10-XX|09:29:35| SHELL: mount
2016-10-XX|09:29:36| SHELL: cat /proc/cpuinfo
2016-10-XX|09:29:41| SHELL: cd /
2016-10-XX|09:29:41| SHELL: rm nyadrop
2016-10-XX|09:29:41| SHELL: rm nya
2016-10-XX|09:29:41| SHELL: echo -n -e '¥x7F¥x45¥x4C¥x46¥x1¥x2¥x1¥x0¥x0¥x0¥x0¥x0¥x0¥[REDACTED]' >> nyadrop // ELF malware

软件调试是如何开始

XiaoYI Ants unofficial info page

HOME INSTRUCTIONS FIRMWARES BUY A YI

Firmwares

Hardware version v2.1 needs a firmware version 1.8.5.1K or higher!
You can find the how to on the firmware flash instruction page.
Note: flash firmware is at your own risk!

Original for CN hardware

- 1.8.5.1B_201513211614
- 1.8.5.1H_201505211709
- 1.8.5.1J_201507201424
- 1.8.5.1K_201508311131
- 1.8.5.1L_201506291725
- 1.8.5.1M_201512011815
- 1.8.5.1N_201512212009
- 1.8.6.1A_201602241619
- 1.8.6.1B_201603181307

Original for international hardware

- 1.8.5.1N_201601071352

Modified for CN hardware

Additional features are added to this firmwares (RTSP, FTP, telnet, timezone, ...)
How to use the different additional features is described on the instruction page.

- 1.8.5.1B_rtsp
- 1.8.5.1J_easy_boot
- 1.8.5.1K_rtspfix-v3
- 1.8.5.1L_rtspfix-v3
- 1.8.5.1M_rtspfix-v4
- 1.8.6.1B_rtspfix

软件调试

- **下载固件**
- **拆开包装**
- **分析web**
- **分析二进度**
- **寻找漏洞**
- **漏洞**

安全科技公司

软件分析知识

```
root@kali:~/iot/firmware-mod-kit/fmk/rootfs/usr/bin# ls -al
total 28
drwxr-xr-x 2 root root 4096 Jan 21  2015 .
drwxr-xr-x 6 root root 4096 Mar 27  2014 ..
lrwxrwxrwx 1 root root   17 Jun 30 05:48 [ -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 [[ -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 arping -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 awk -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 basename -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 bunzip2 -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 bzcat -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 clear -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 cmp -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 crontab -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 cut -> ../../bin/busybox
lrwxrwxrwx 1 root root   16 Jun 30 05:48 dbclient -> ../sbin/dropbear
lrwxrwxrwx 1 root root   17 Jun 30 05:48 dirname -> ../../bin/busybox
lrwxrwxrwx 1 root root   16 Jun 30 05:48 dropbearkey -> ../sbin/dropbear
lrwxrwxrwx 1 root root   17 Jun 30 05:48 du -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 env -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 expr -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 find -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 free -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 head -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 hexdump -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 hostid -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 id -> ../../bin/busybox
-rwxr-xr-x 1 root root 7144 Mar 27  2014 jshn
lrwxrwxrwx 1 root root   17 Jun 30 05:48 killall -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 less -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 logger -> ../../bin/busybox
-rwxr-xr-x 1 root root 9284 Mar 27  2014 lua
lrwxrwxrwx 1 root root   17 Jun 30 05:48 md5sum -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 mkfifo -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 nc -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 nslookup -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 passwd -> ../../bin/busybox
lrwxrwxrwx 1 root root   17 Jun 30 05:48 pgrep -> ../../bin/busybox
```

软件调试

- LINUX
- Web
- Perl
- Python
- 汇编
- ARM
- MIPS

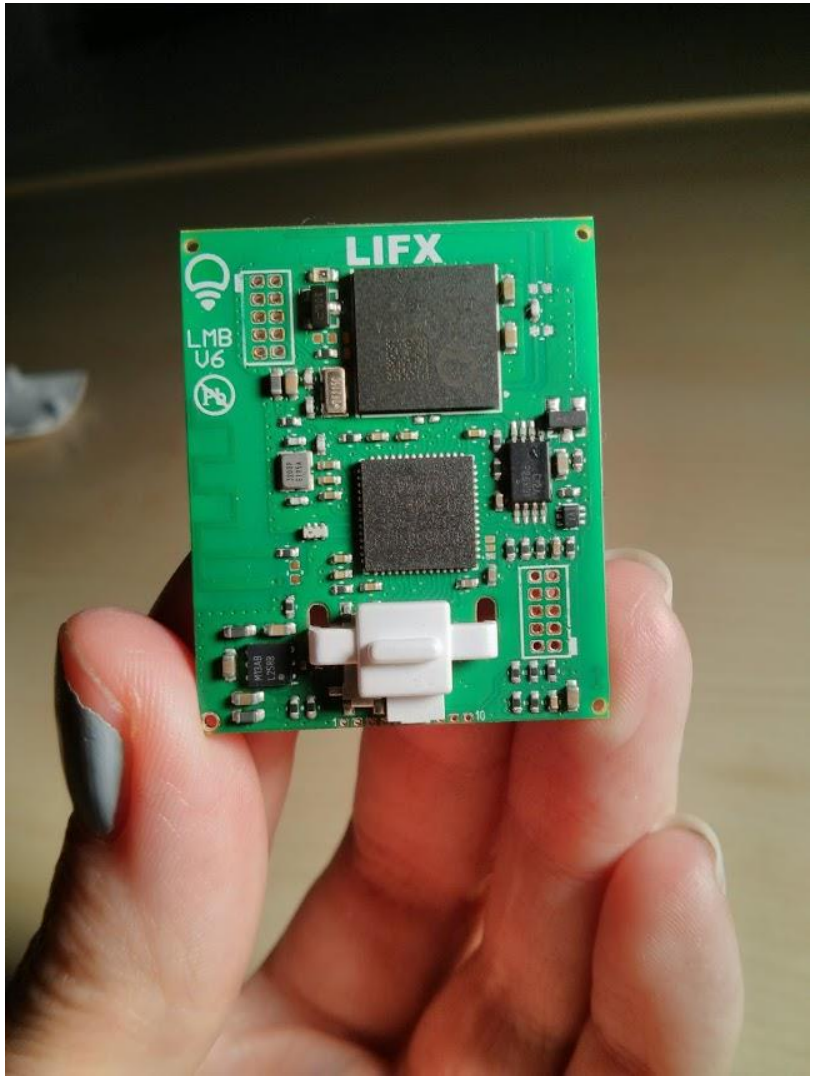
安全科技公司

厂家措施

升级过程解密


Free ZIP Password Cracker
Free ZIP Password Cracker
Passwords
test.exe = 908365
Control
Archive File:
lsers\user99\Desktop\setup.zip
...
Starting Password:
908365
FileSpec:
*.*exe
CharType Property
Numeric (0-9)
Alpha (upper) (A-Z)
Alpha (lower) (a-z)
Alpha (upper/lower case) (A-Z, a-z)
AlphaNumeric (upper/lower case) (0-9, A-Z, a-z)
AlphaNumeric (upper case) (0-9, A-Z)
AlphaNumeric (lower case) (0-9, a-z)
Full ascii set
NonAlphaNumeric
WordList
\\192.168.1.109\english data\Ne
...
Info
Current File:
test.exe
Password hits:
1049810
Forced hits:
4138
Password Length:
6
Load Last Saved Password
Shutdown when finished
Start
Pause
Display Off
Help

固件加密



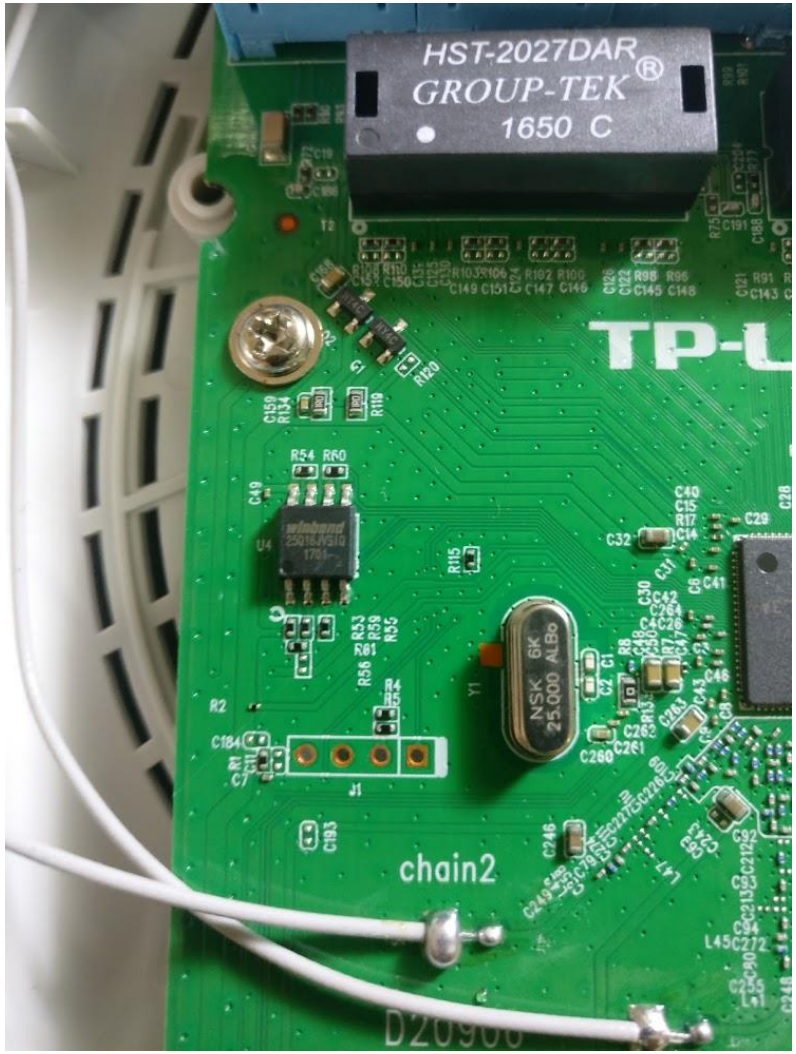

```
MOVS     R2, #0x68          ; Rd = Op2
MOV      R0, R4             ; Rd = Op2
LDR      R1, =0x20003AA0    ; Load from Memory
BL       sub_8035060        ; Branch with Link
ADD      R0, SP, #0x330+var_128 ; Rd = Op1 + Op2
MOVS     R2, #0x80          ; Rd = Op2
LDR      R1, =AES_Key       ;
BL       Ref_sbox_1         ; Branch with Link
ADD      R0, SP, #0x330+var_128 ; Rd = Op1 + Op2
MOVS     R1, #1             ; Rd = Op2
MOVS     R2, #0x70          ; Rd = Op2
LDR      R3, =AES_IV        ; Load from Memory
STMEA.W  SP, {R4,R5}        ; Store Block to Memory
BL       Ref_Ref_sbox       ; Branch with Link
MOVS     R0, #0x70          ; Rd = Op2
ADD.W    SP, SP, #0x324     ; Rd = Op1 + Op2
POP      {R4,R5,PC}         ; Pop registers
; End of function AES
```

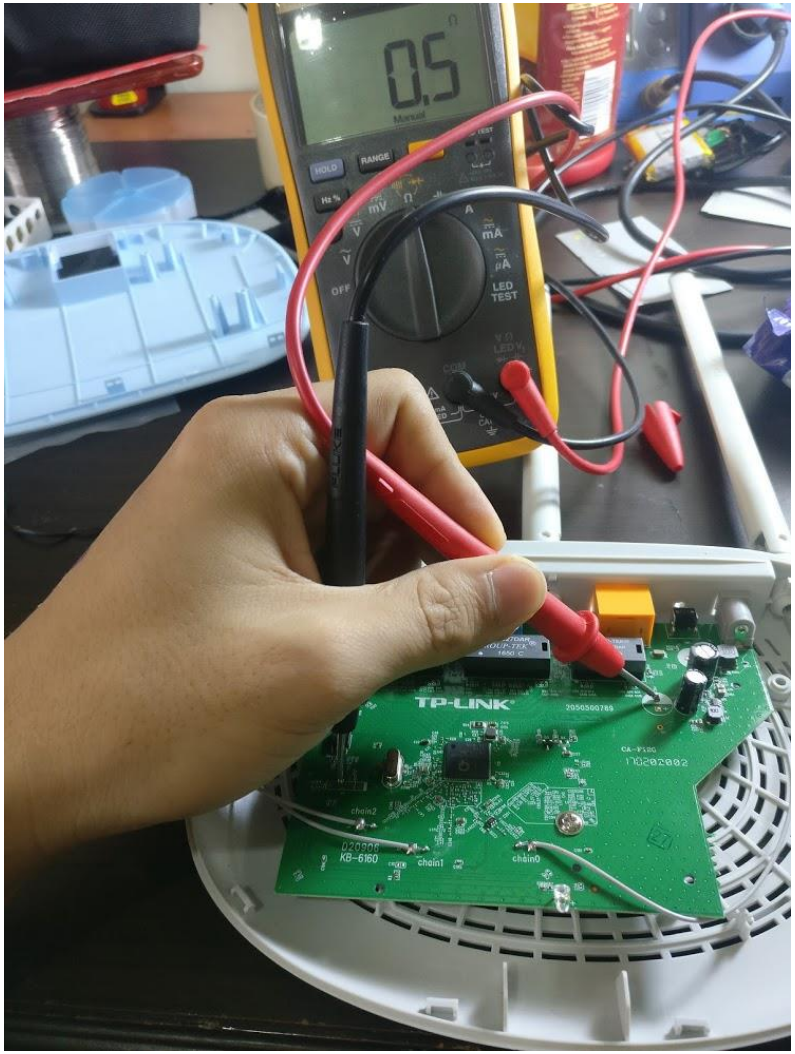
硬件分析

拆机

UART

寻找GND





课程目的

最短的时间学会黑


Version2.0
Software Platform for PNE2.2
Copyright(C) 2001-2011 by TP-LINK TECHNOLOGIES CO., LTD.
Creation date: Nov 15 2016, 11:52:02
Install minifs driver successed.
MAC address B8F8839B 0C850000
B8 F8 83 9B 0C 85
0xEF4015.
0xE666909919F513624
MAC address B8F8839B 0C860000
B8 F8 83 9B 0C 86
link mode 0x00.
default gateway 169.254.52.97
net 169.254.52.0
net 0.0.0.0
WLAN event task tWLANEventTask
te mem partion 0x809137f0 for size 94464
terprise mode: 0x03bda000
storing Cal data from Flash
ase_eep_header.feature_enable = 0x7d.
ase_eep_header.txrxgain = 0x21.
odal_header_2g.temp_slope = 0x12.
odal_header_5g.temp_slope = 0x44.
base_ext2.temp_slope_low = 0x12.
base_ext2.temp_slope_high = 0x12.
Create OS_MESGQ [scan] Successfully at 0x808d9320
osifp->queue_head = 0x808d9678
Create OS_MESGQ [pwrsav_sta] Successfully at 0x80606900
Create OS_MESGQ [connection] Successfully at 0x80600780
Create OS_MESGQ [assoc] Successfully at 0x80600460
Entering WLAN event Task Loop
ENG
INTL
1:03 AM
5/25/2017
Delete

工具

入门

买1送5 胜利正品数字万用表VC890C+ 全保护万能表数显多用表电表

胜利经典款 欧洲安全标准 测试快稳定

天猫 购物券 全天猫实物商品通用 去领券

价格 ~~¥176.00-612.00~~

促销价 **¥88.00-306.00**

本店活动 满2件9.8折；满5件9.6折 更多优惠

运费 湖南长沙 至 杭州 快递: 0.00 EMS: 25.00 平邮: 30.00

月销量 4691 | 累计评价 35418 | 送天猫积分 44起

颜色分类

- VC890C+标配【送鳄鱼夹和仪表包】
- VC890C+标配+仪表包【送鳄鱼夹】
- VC890C+标配【送鳄鱼夹】
- VC890C+标配+20A原装表笔+充电套装【送鳄鱼夹】
- VC890C+标配+充电套装【送鳄鱼夹】
- VC890C+标配+仪表包+充电套装【送鳄鱼夹】
- VC890C+标配+仪表包+20A原装表笔【送鳄鱼夹】
- VC890C+标配+20A特尖+充电套装【送鳄鱼夹】
- VC890C+标配+20A原装表笔【送鳄鱼夹】

分享 收藏商品（20733人气） 举报

单排针 圆排针2.54MM 2.0MM间距1*40P双排针2*40P 弯针单双排针

品质保证 长久耐用 7天包退换

天猫 购物券 全天猫实物商品通用 去领券

价格 **¥2.50**

运费 广东深圳 至 广州 快递: 0.00

月销量 808 | 累计评价 1100

颜色分类

- 2.54单排针 铁【普通长度】 10条
- 2.54单排针 铜【长15MM】10条
- 2.54单排针 铜【长17MM】10条
- 2.54单排针 铜【长19MM】 10条
- 2.54单排针 铜【长25MM】10条
- 2.54双排针 铜【长15MM】10条
- 2.54单排弯针 铜【长11MM】 10条
- 2.54双排弯针 铜 【长11MM】 10条
- 2.54双排针 铜【长11MM】10条
- 2.54双排针 铜【长19MM】10条
- 2.54双排针 铜【长25MM】10条
- 2.0单排针 铜 10条
- 2.0单排弯针 铜 10条
- 2.54镀金单排针 10条

Risym 六合一多功能USB转UART串口模块CP2102 usb/TTL485/232互转

RISYM 出口 质优产品

天猫 购物券 全天猫实物商品通用 去领券

价格 ~~¥19.90~~

促销价 **¥16.80** 首件体验价

本店活动 满68元,包邮；满688元,包邮,赠满688送充电宝 更多优惠

运费 广东深圳 至 广州 快递: 3.00 EMS: 23.00

月销量 120 | 累计评价 147 | 送天猫积分 1

数量 1 件 库存8954件

立即购买 加入购物车

服务承诺 正品保证 极速退款 七天无理由退换 支付方式

QUICK快克203H/203D数显无铅高频恒温焊台90W大功率烙铁204电焊台

控温准确 回温讯速 大功率90W

天猫 购物券 全天猫实物商品通用 去领券

价格 ~~¥888.00~~

促销价 **¥850.00** 品牌钜惠

运费 广东深圳 至 广州 快递: 12.00 EMS: 70.00 平邮: 39.00

月销量 49 | 累计评价 179 | 送天猫积分 425

颜色分类

- 203H(数显90W)
- 204H(机械90W)
- 203(数显60W)
- 204(机械60W)
- 203D(双数显90W)

数量 1 件 库存586件

立即购买 加入购物车

服务承诺 正品保证 七天无理由退换 支付方式

分享 收藏商品（376人气） 举报

注意事项 A

分享 ★ 收藏商品（20733人气） 举报

买1送5 胜利正品数字万用表VC890C+ 全保护万能表数显多用表电表

胜利经典款 欧洲安全标准 测试快稳定

天猫 购物券 全天猫实物商品通用 去刮券

价格 ~~¥176.00-612.00~~

促销价 ¥88.00-306.00

本店活动 满2件9.8折；满5件9.6折 更多优惠

运费 湖南长沙 至 杭州 快递: 0.00 EMS: 25.00 平邮: 30.00

月销量 4691 | 累计评价 35418 | 送天猫积分 44起

颜色分类

- VC990C+标配【送鳄鱼夹和仪表包】
- VC890C+标配+仪表包【送鳄鱼夹】
- VC890C+标配【送鳄鱼夹】
- VC890C+标配+20A原装表笔+充电套装【送鳄鱼夹】
- VC890C+标配+充电套装【送鳄鱼夹】
- VC890C+标配+仪表包+充电套装【送鳄鱼夹】
- VC890C+标配+仪表包+20A原装表笔【送鳄鱼夹】
- VC890C+标配+20A特尖+充电套装【送鳄鱼夹】
- VC890C+标配+20A原装表笔【送鳄鱼夹】

分享 ★ 收藏商品（11人气） 举报

福禄克万用表F15B+数字万用表FLUKE17B+/18B+高精度数字万能表

原装正品 新升级

天猫 购物券 全天猫实物商品通用 去刮券

价格 ~~¥499.00-699.00~~

促销价 ¥468.00-684.00

运费 广东东莞 至 杭州 快递: 0.00

月销量 11 | 累计评价 10 | 送天猫积分 234起

颜色分类

数量 1 件 库存256件

立即购买 加入购物车

服务承诺 正品保证 赠运费险 七天无理由退换 支付方式

- 价格
- 实际分别？

注意事项 B

Risym 六合一多功能USB转UART串口模块CP2102 usb/TTL485/232互转

RISYM 出口 质优产品

全天猫实物商品通用 去刮券

价格 ¥ 49.90

促销价 ¥16.80 首件体验价

本店活动 满68元,包邮；满688元,包邮,赠:满688送充电宝 更多优惠

运费 广东深圳 至 广州 快递: 3.00 EMS: 23.00

月销量 120 累计评价 147 送天猫积分 1

数量 1 件 库存8954件

立即购买 加入购物车

服务承诺 正品保证 极速退款 七天无理由退换 支付方式

分享 收藏商品（25人气） 举报

微雪 FT232模块 FT232RL USB转TTL USB转UART 串口模块 micro接口

FT232 模块 刷机神器

全天猫实物商品通用 去刮券

价格 ¥28.00

本店活动 满508元减10元 更多优惠

运费 广东深圳 至 广州 快递: 8.00 EMS: 24.00

月销量 103 累计评价 50 送天猫积分 2

数量 1 件 库存7350件

立即购买 加入购物车

服务承诺 正品保证 极速退款 赠运费险 七天无理由退换 支付方式

¥3.10

40P彩色杜邦线 母对母 公对公 公对母

荣域华府旗舰店

月成交 2474笔 评价 9561

¥6.19

域能 40P彩色杜邦线母对母 公对公 公

域能旗舰店

月成交 360笔 评价 2251

¥3.28

40P彩色杜邦彩虹排线电子延长线 公转

艾速旗舰店

月成交 500笔 评价 1418

- 寻找两个分别

注意事项 C

单排针 圆排针2.54MM 2.0MM间距1*40P双排针2*40P 弯针单双排针

品质保证 长久耐用 7天包退换

天猫 购物券 全天猫实物商品通用 去刮券

价格 ¥2.50

运费 广东深圳 至 广州 快递: 0.00

月销量 808 累计评价 1100

颜色分类

2.54单排针 铁【普通长度】 10条
2.54单排针 铜【长15MM】 10条
2.54单排针 铜【长17MM】 10条
2.54单排针 铜【长19MM】 10条
2.54单排针 铜【长25MM】 10条
2.54双排针 铜【长15MM】 10条
2.54单排弯针 铜【长11MM】 10条
2.54双排弯针 铜 【长11MM】 10条
2.54双排针 铜【长11MM】 10条
2.54双排针 铜【长19MM】 10条
2.54双排针 铜【长25MM】 10条
2.0单排针 铜 10条
2.0单排弯针 铜 10条
2.54镀金单排针 10条

ARTHYLY 间距2.00MM 双排弯针 双排针 弯 插针 2*40PIN

天猫 购物券 全天猫实物商品通用 去刮券

价格 ¥2.00

促销价 ¥1.98 促销

运费 广东深圳 至 广州 快递: 0.00 EMS: 0.00 平邮: 0.00

月销量 10 累计评价 5

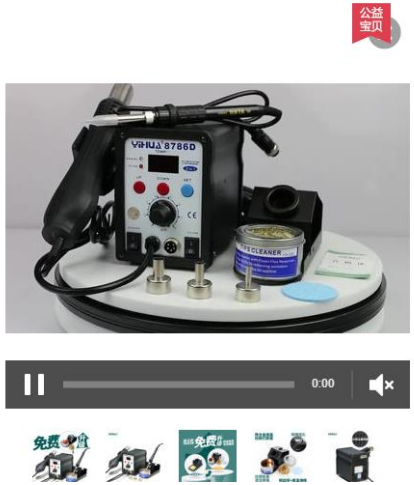
数量 1 件 库存189599件

立即购买 加入购物车

服务承诺 正品保证 极速退款 七天无理由退换 支付方式

- 寻找两个分别
- 2.0 / 2.54
- 直的 还是 L的

注意事项 D



- Made in china

其他工具

- 告诉我4个的用处

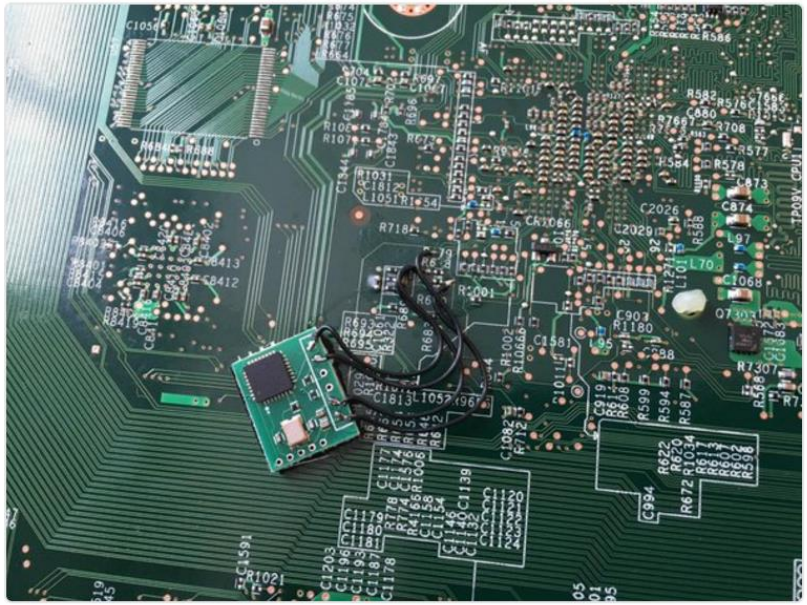
目标

终极目标

You Retweeted

Hacker Fantastic @hackerfantastic · 13 May 2016

Suspicious daughterboard inside Chinese sold cisco gear via @osxreverser ...
soooo dodgy lol



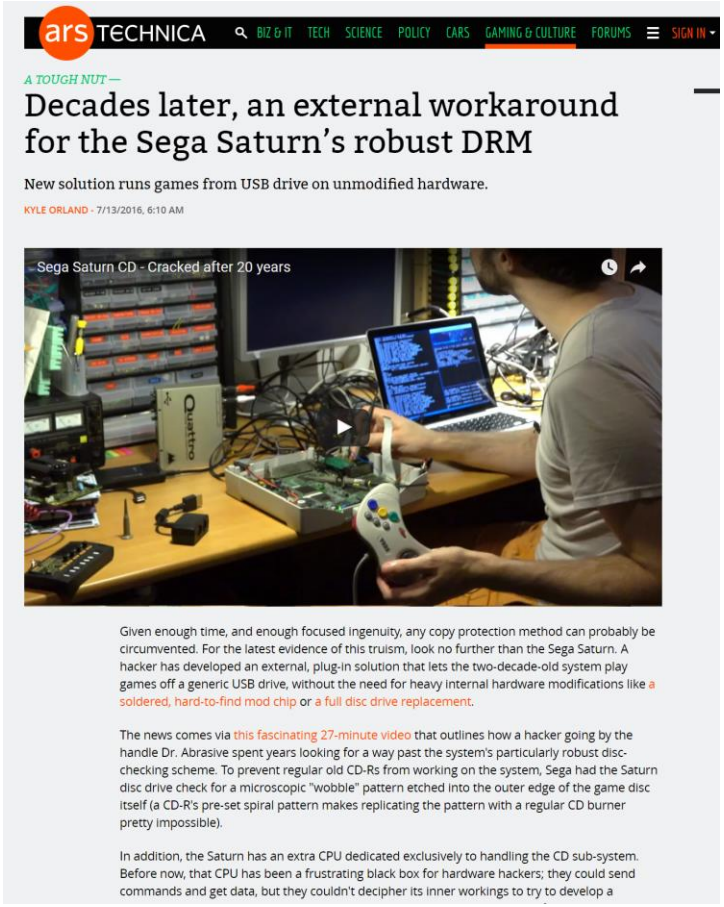
34 459 369

- 你买的东西是不是你要的东西
- 黑的过程学到了什么

攻击无人机

- 除了IOT 无人机也是一个好的学习目标
- SDR + 固件 + 硬件分析

翻新



ars TECHNICA BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE FORUMS SIGN IN

A TOUGH NUT —

Decades later, an external workaround for the Sega Saturn's robust DRM

New solution runs games from USB drive on unmodified hardware.

KYLE ORLAND - 7/13/2016, 6:10 AM

Given enough time, and enough focused ingenuity, any copy protection method can probably be circumvented. For the latest evidence of this truism, look no further than the Sega Saturn. A hacker has developed an external, plug-in solution that lets the two-decade-old system play games off a generic USB drive, without the need for heavy internal hardware modifications like a soldered, hard-to-find mod chip or a full disc drive replacement.

The news comes via this fascinating 27-minute video that outlines how a hacker going by the handle Dr. Abrasive spent years looking for a way past the system's particularly robust disc-checking scheme. To prevent regular old CD-Rs from working on the system, Sega had the Saturn disc drive check for a microscopic "wobble" pattern etched into the outer edge of the game disc itself (a CD-R's pre-set spiral pattern makes replicating the pattern with a regular CD burner pretty impossible).

In addition, the Saturn has an extra CPU dedicated exclusively to handling the CD sub-system. Before now, that CPU has been a frustrating black box for hardware hackers; they could send commands and get data, but they couldn't decipher its inner workings to try to develop a

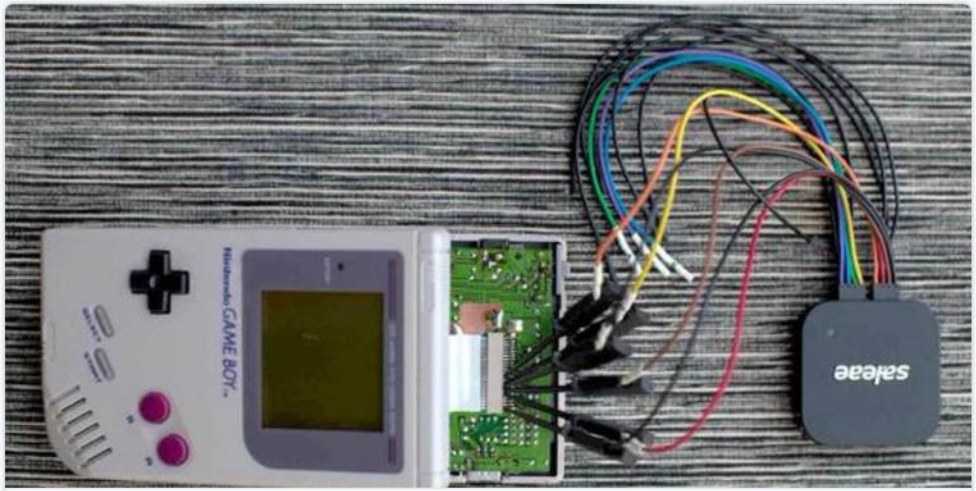
- 老玩具也是玩具
- 如何将SEGA SATURN改装成为U盘

翻新

You Retweeted

hackaday @hackaday · Aug 2

Using a Logic Analyzer to Generate Screenshots from a Game Boy



Using a Logic Analyzer to Generate Screenshots from a Game Boy

Wouldn't you like to go back to a dead handheld and extract the proof of your 90s-era high scores? Of course you would. [svendahlstrand] bought ...

hackaday.com

- Gameboy 截屏

翻新

- 车子也是可以的
- 尤其是有带网络的车子

小工具

ESP8266串口wifi模块 NodeMcu Lua WIFI V3 物联网 开发板 CH-340

价格 ￥16.30 346 累计评论 290 交易成功

优惠 淘金币可抵0.16元

店铺 优惠券 10元店铺优惠券，满398元可用 领取

店铺 优惠券 5元店铺优惠券，满168元可用 领取

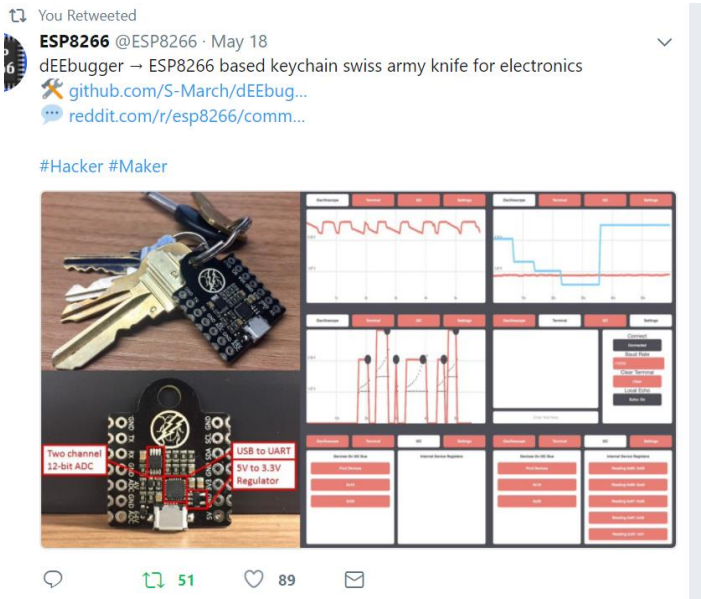
配送 广东深圳 至 广东广州白云区 快递 ¥3.00

数量 1 件(库存16565件)

立即购买 加入购物车

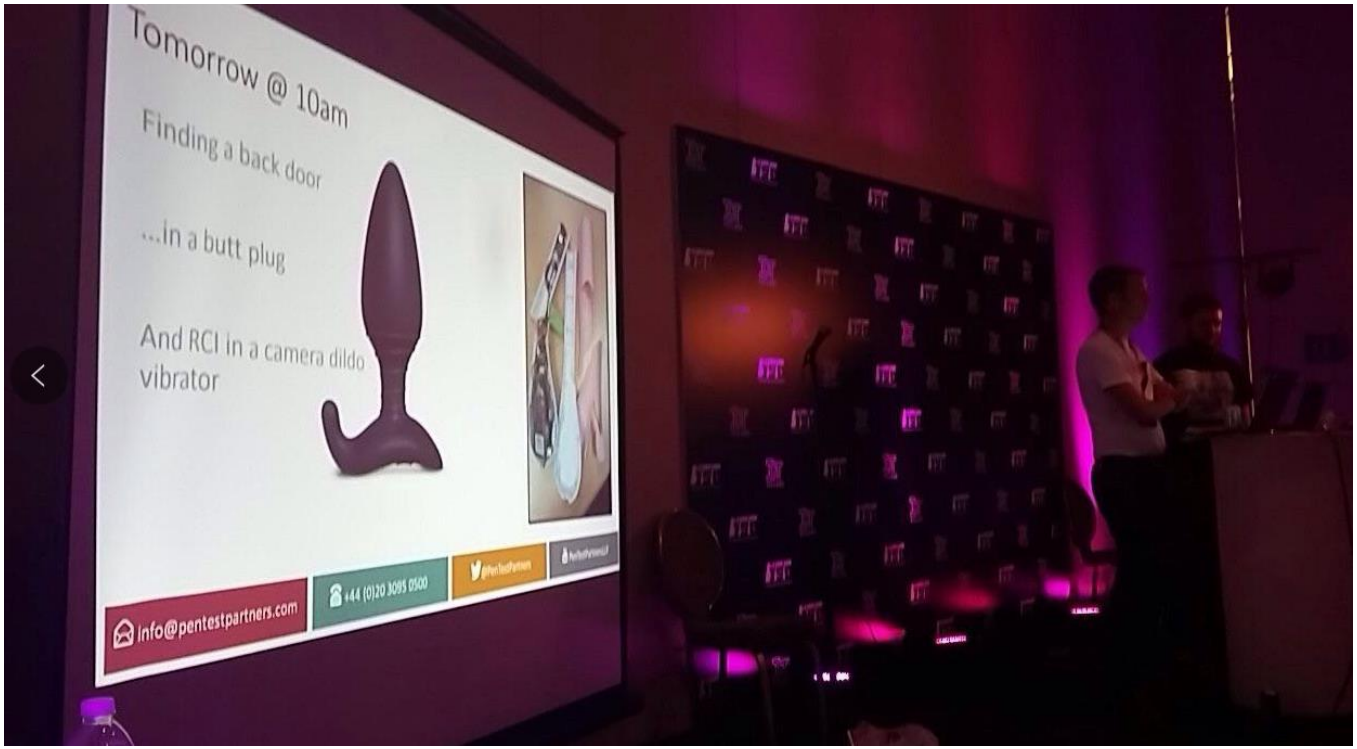
承诺 7天无理由

支付 蚂蚁花呗 信用卡支付 集分宝



- 一些好玩的小工具
- https://github.com/S-March/dEEbugger_PUBLIC

学习英文的时候



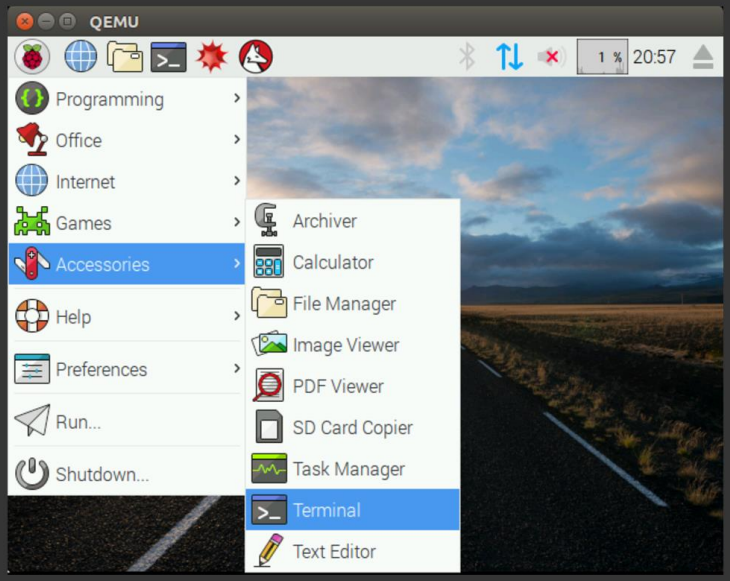
- 讨论时间


需要买很多硬件吗

QEMU

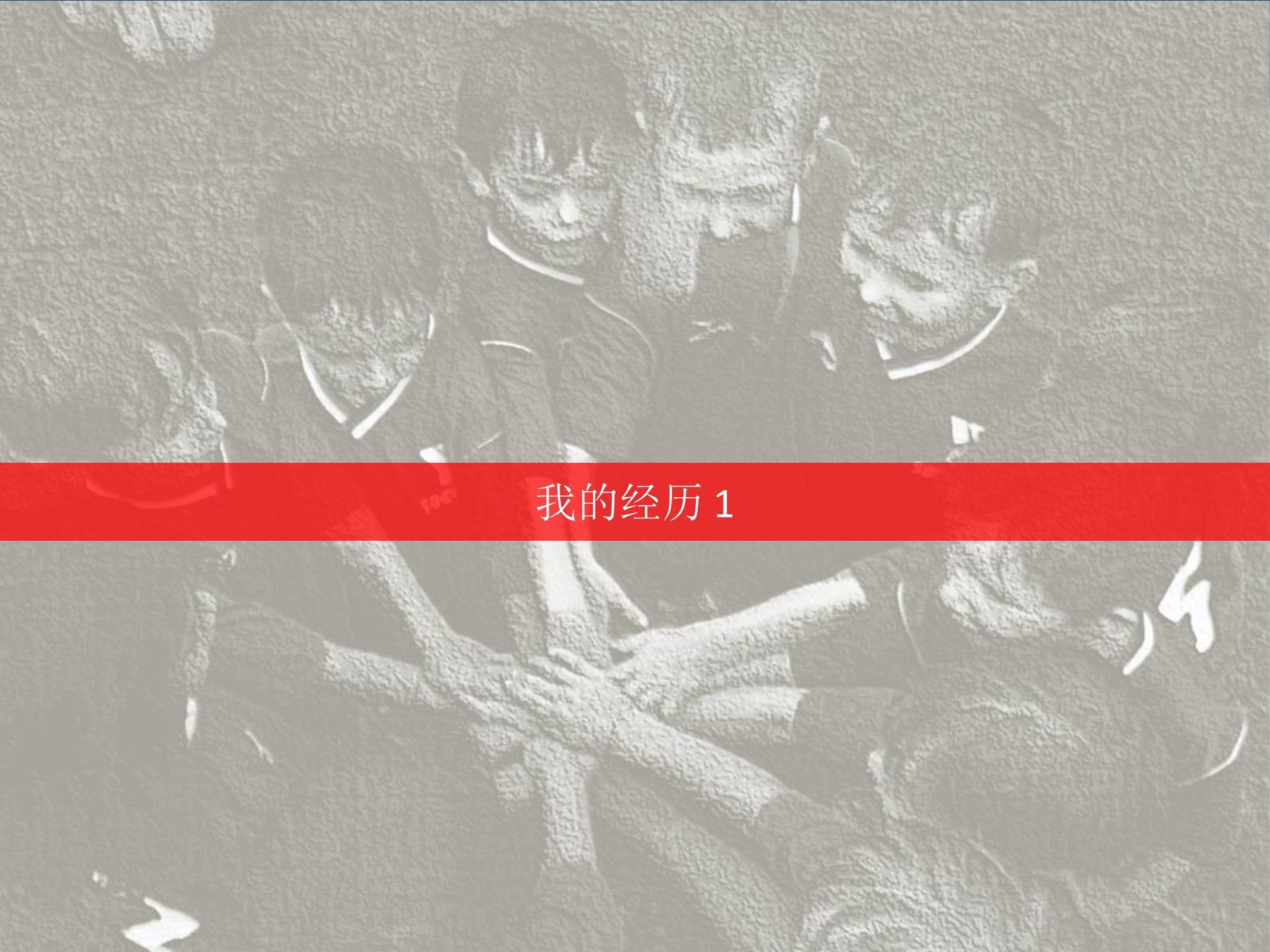
```
$ qemu-system-arm -kernel ~/qemu_vms/<your-kernel-qemu> -cpu arm1176 -m 256 -M versatilepb -serial stdio -append
```

If you see GUI of the Raspbian OS, you need to get into the terminal. Use Win key to get the menu, then navigate with arrow keys until you find Terminal application as shown below.



From the terminal, you need to start the SSH service so that you can access it from your host system (the one from which you launched the qemu).

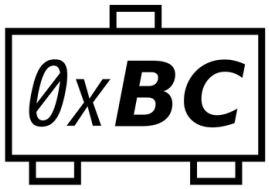
- 硬件是一个必须品吗



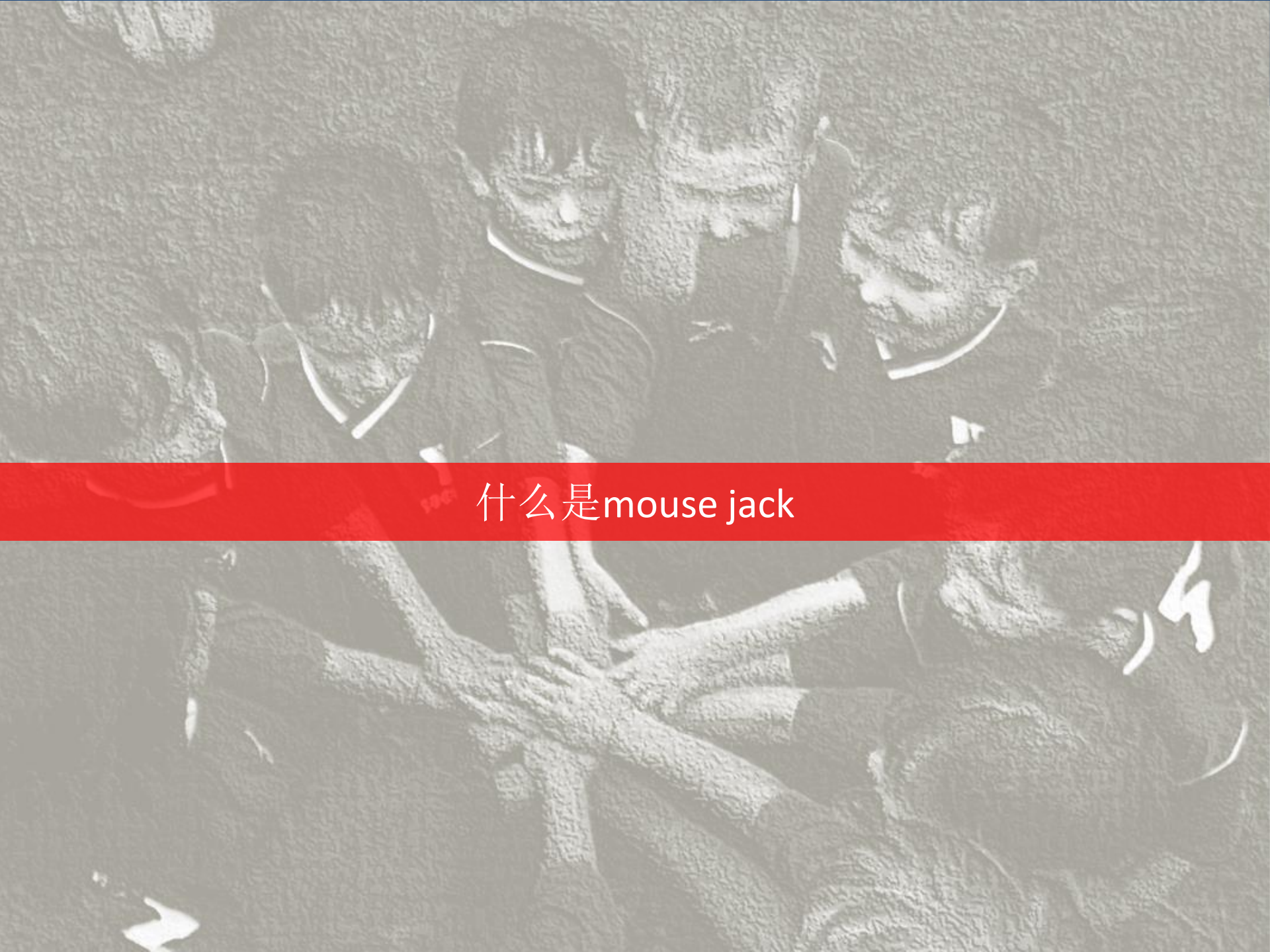
我的经历 1

Research Credits

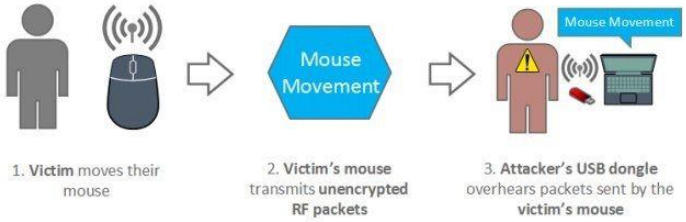
黑客与极客



- Partner In Crime: klks84, https://twitter.com/klks84
- All my missing Logitech keyboard and mouse

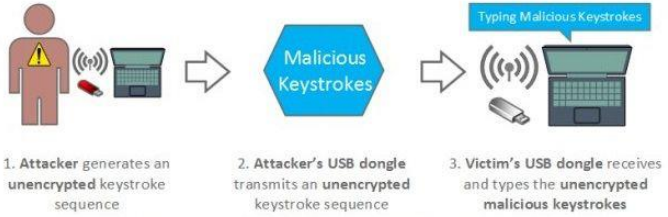

什么是mouse jack

MouseJack



Attacker Identifying a Victim's Mouse or Keyboard

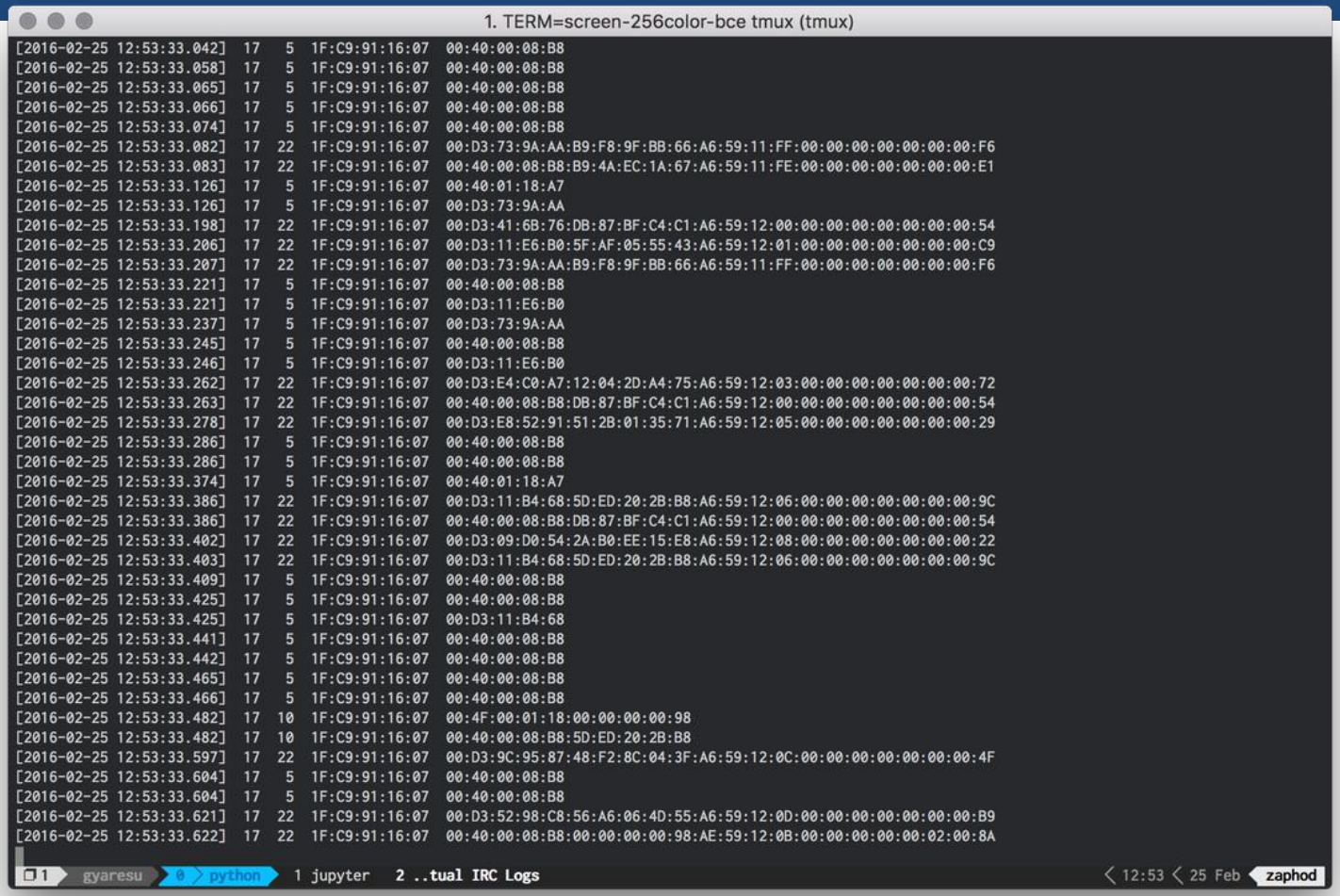
Attacker Force-Pairing a Fake Keyboard with the Victim's Dongle



Attacker Injecting Keystrokes into the Victim's Dongle

- Targeting non-Bluetooth keyboard and mice
- Sniff and transmit special crated radio packet towards victims
- Keyboards normally sends encrypted packets
- Affected Product ? Most of the non-Bluetooth keyboard and mouse

How It Works



- Scan all the nearby wireless mouse
- Sniff targeted victim
- Dump “keystroke”
- Replay, Hijack, 0w3d

目的

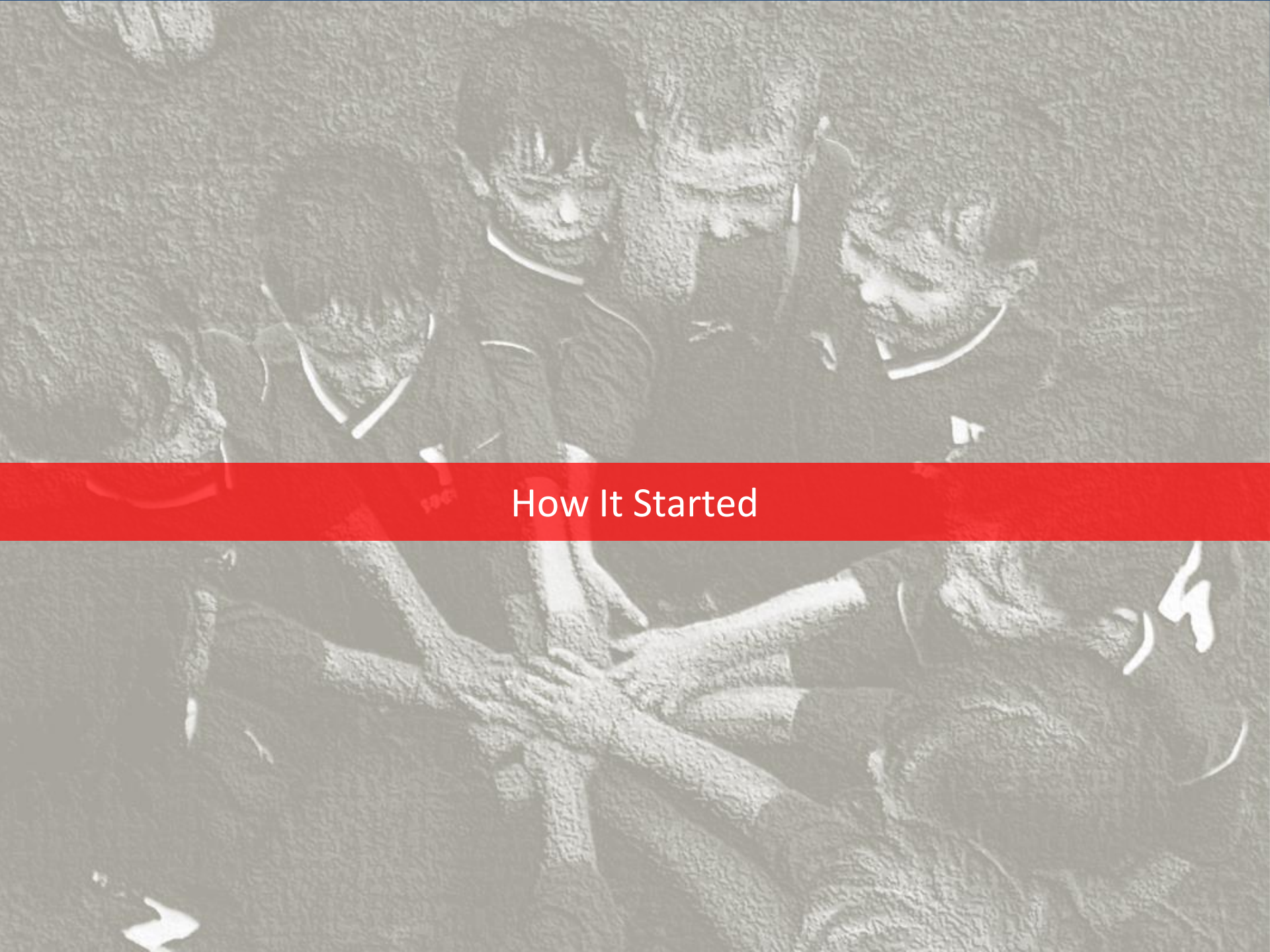
Why This Research

- Most complete MouseJack implementation guide, in chinese
- Both guide based on Crazyradio. “PA” and non “PA”

- Objective 1: Can it be cheaper?
- Objective 2: Smaller? (Not too obvious)
- Objective 3: Easier to purchase? Just tabao it?

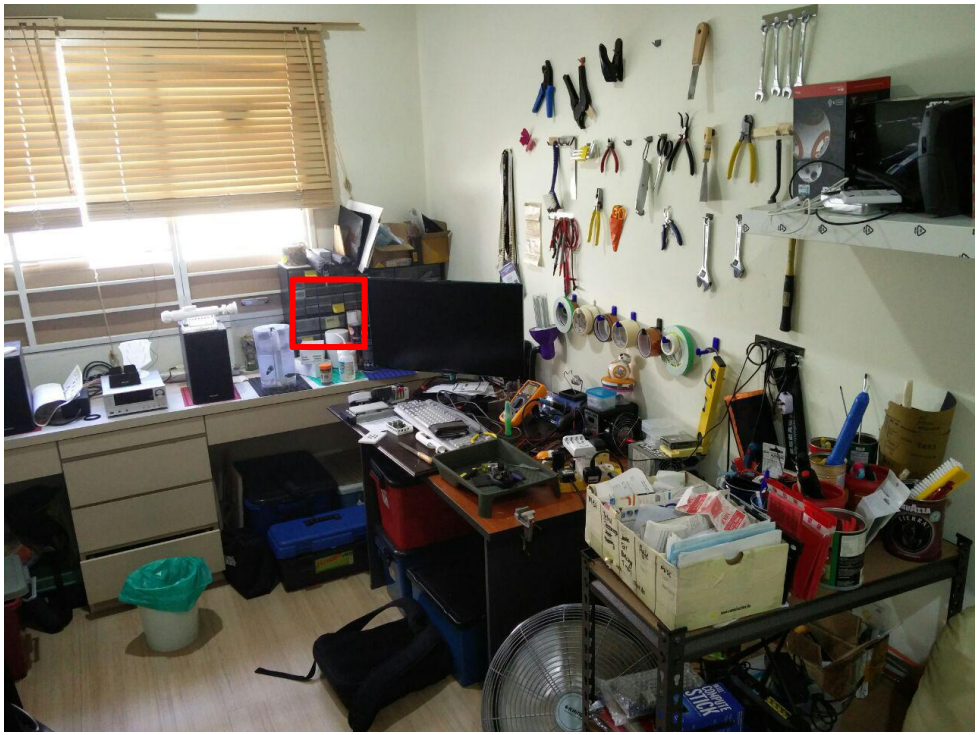
What Not

- Nothing to do with keyboard direct injection
- Nothing to do with breaking keyboard encryption
- Nothing to do with mouse injection
- Nothing to do with super long distance sniffing
- BUT



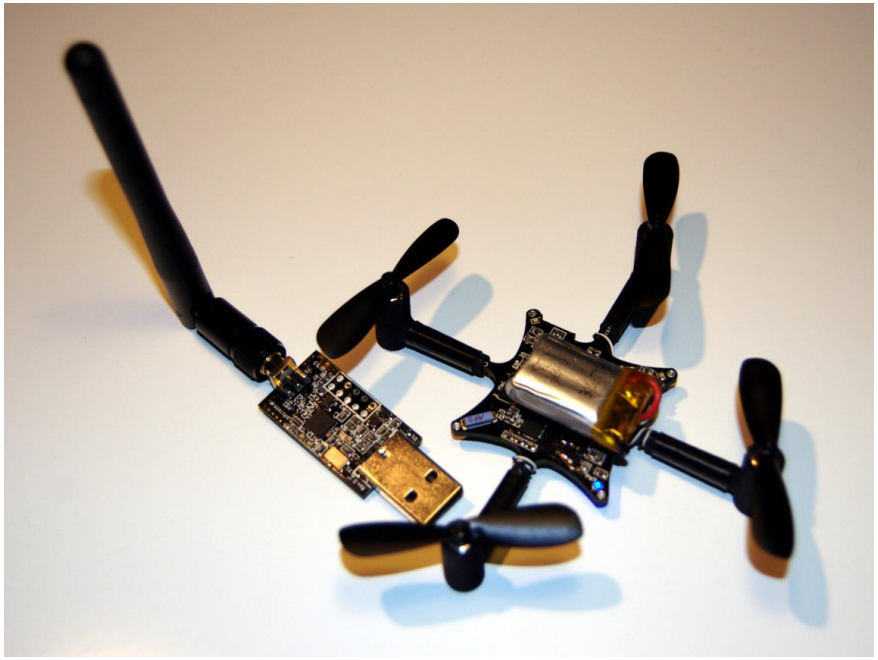
How It Started

It Does Ring the Bell



- Got it few years back
- Always hide in small little corner
- After two times flying
- I think I give up, completely

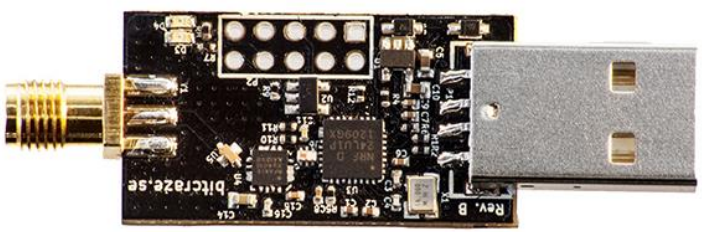
The Crazyradio

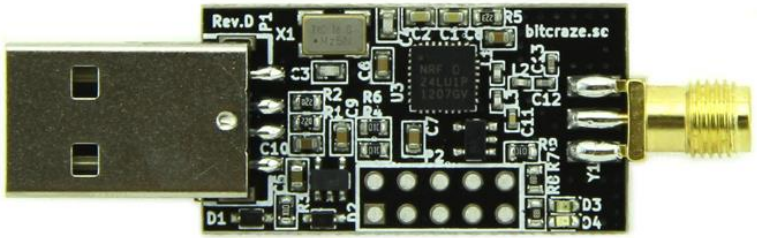


- Based on nRF24LU1+ chip
- 125 radio channels
- Send and receive packets up to 32 bytes
- Design by bitcraze.io to fly crazyflies
- Comment: Not too easy to fly


I Kill It

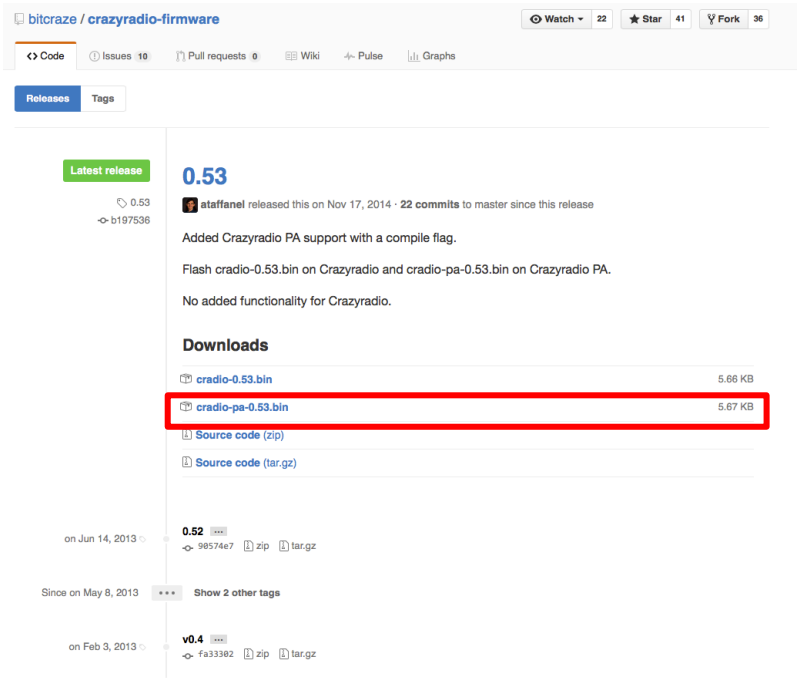
Two Different Types, PA and Non PA

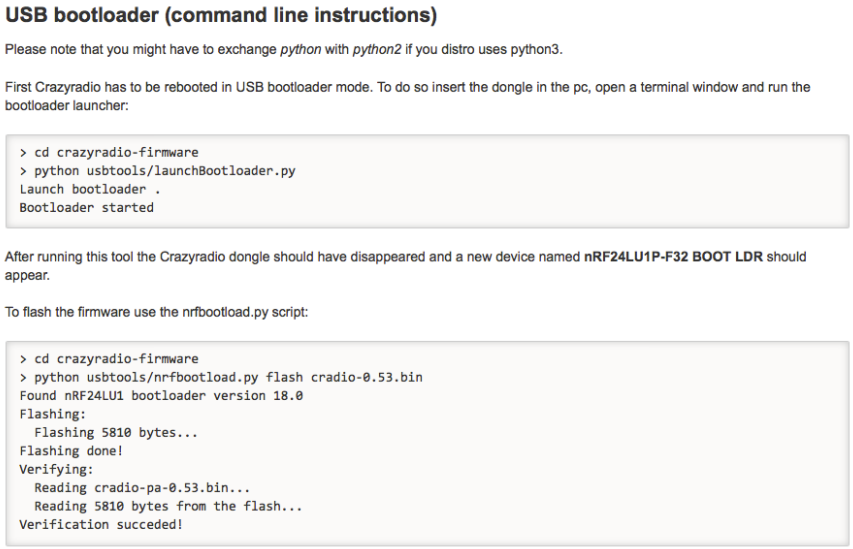




- In the beginning, there are two types of crazyradio
- Item 1. Crazyradio PA
- Item 2, Crazyradio (Obsolete)
- Crazyradio PA comes with extended range. 1KM
- Bitcraze no longer selling crazyradio, only PA model is available

How It All Started





USB bootloader (command line instructions)

Please note that you might have to exchange *python* with *python2* if you distro uses python3.

First Crazyradio has to be rebooted in USB bootloader mode. To do so insert the dongle in the pc, open a terminal window and run the bootloader launcher:

```
> cd crazyradio-firmware
> python usbtools/launchBootloader.py
Launch bootloader .
Bootloader started
```

After running this tool the Crazyradio dongle should have disappeared and a new device named **nRF24LU1P-F32 BOOT LDR** should appear.

To flash the firmware use the nrfbootload.py script:

```
> cd crazyradio-firmware
> python usbtools/nrfbootload.py flash cradio-0.53.bin
Found nRF24LU1 bootloader version 18.0
Flashing:
  Flashing 5810 bytes...
Flashing done!
Verifying:
  Reading cradio-pa-0.53.bin...
  Reading 5810 bytes from the flash...
Verification succeded!
```

- At the beginning, there are two crazyradio
- Flashing the “PA” firmware in to the the NON “PA” is a bad idea
- Somehow, boot loader been overwritten
- The End

Solutions

Using BusPirate

It's possible to re-program the Crazyradio using a BusPirate and this script via SPI.

Couple of caveats:

- Tested only on OS X. Should work on Linux without modification, and Windows with very minor changes to use the windows serial module.
- It's very slow (~5 minutes to flash the entire .bin file). I deemed this acceptable as this script is for emergency recovery only. I can make it faster if necessary.

Prerequisites:

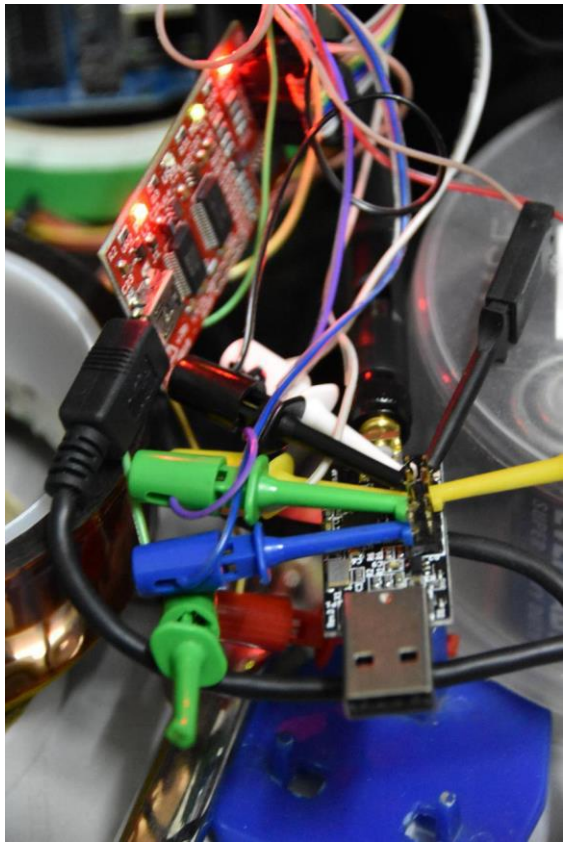
- A Bus Pirate (you should know where to get one of these, if you don't already have one).
- perl and either Device::SerialPort (*nix) or Win32::SerialPort (Windows)
- Some jumper wires to connect the SPI lines on the radio to the ones on the Bus Pirate.

Instructions:

1. Solder a 2×5 pin header onto the programming port of the crazyradio. There's an unpopulated footprint already there for you.
2. Connect the crazyradio to your Bus Pirate using the table below (also noted in the script and readme on git)

```
Bus Pirate       CrazyRadio
==========================
MOSI ()      ->  MOSI  (6)
MISO ()      ->  MISO  (8)
SCK  ()      ->  SCK   (4)
CS   ()      ->  CS    (10)
AUX  ()      ->  PROG  (2)
3V3  ()      ->  3V3   (5)
GND  ()      ->  GND   (9)
```

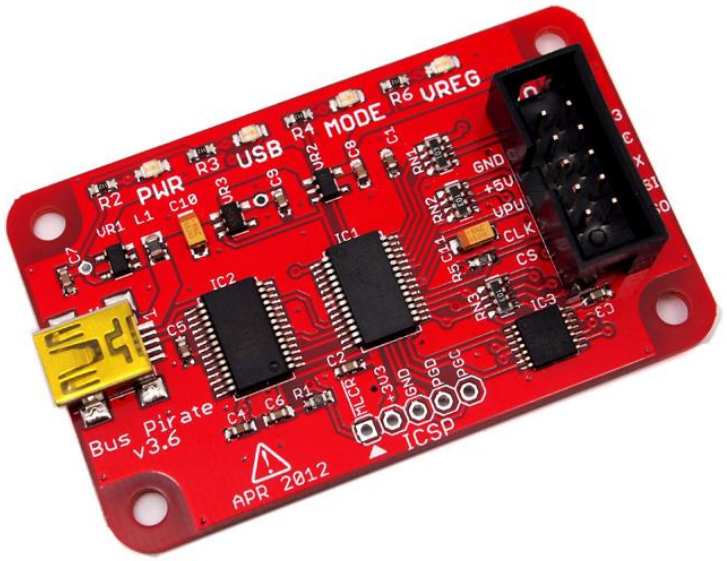
3. Run the script: perl ./flasher.pl -input ./cradio-0.51.bin -device [serial device]
4. Wait till you see lots of hex addresses crawling up your screen. Your device is programming.
5. Go make a sandwich or have a beer (or both).

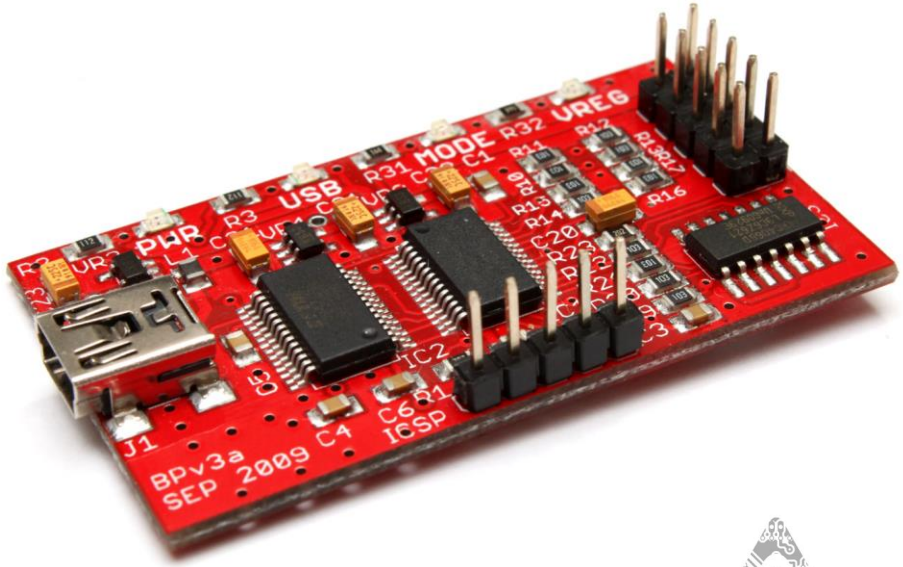


- Crazyradio wiki says need a complete SPI flash
- 1 Unit Bus Pirate
- 1 Unit 2 x 5 Pins
- Almost stable hand

需要的工具

What Is Bus Pirate





Dangerous Prototypes

- Item 1, version 3.6
- Item 2, version 4
- Support 1-Wire, I2C, SPI, JTAG, Asynchronous Serial, MIDI and etc
- SPI is what we need
- Sells by seeeds studio and not in taobao

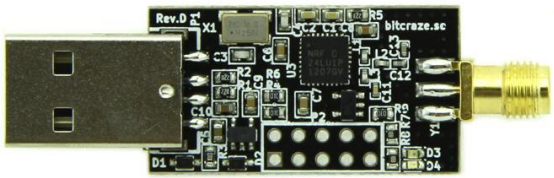
The Perl Script

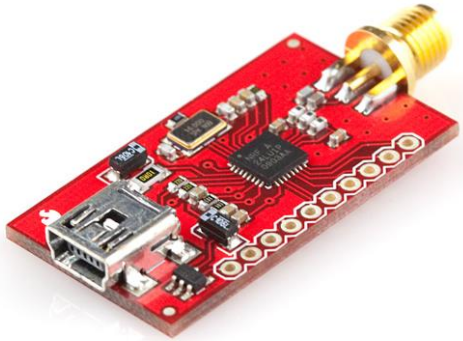
```
#!/usr/bin/perl -w

# Simple perl script to drive the Bus Pirate and unbrick your CrazyRadio dongle.
# Adapted (sorta) from the Bus Pirate example script and mbed NRF24LU1+ flasher projects:
#  http://code.google.com/p/the-bus-pirate/source/browse/trunk/scripts/SPIeeprom.pl
#  http://mbed.org/users/mux/code/nrflash
#
# This script uses the aux output on the Bus Pirate as the PROG pin on the CrazyRadio's NRF24LU1+ chip.
#
# Electrical connections are as follows:
#
#  Bus Pirate       CrazyRadio
#  ==========================
#  MOSI ()      ->  MOSI  (6)
#  MISO ()      ->  MISO  (8)
#  SCK  ()      ->  SCK   (4)
#  CS   ()      ->  CS    (10)
#  AUX  ()      ->  PROG  (2)
#  3V3  ()      ->  3V3   (5)
#  GND  ()      ->  GND   (9)

use strict;
use feature 'say';
use Getopt::Long;
use Device::SerialPort;
use Time::HiRes qw/usleep/;

use constant {
    WREN        => "\x06",
    WRDIS       => "\x04",
    RDSR        => "\x05",
    WRSR        => "\x01",
    READ        => "\x03",
    PROGRAM     => "\x02",
    ERASE_PAGE  => "\x52",
    ERASE_ALL   => "\x62",
    RDFPCR      => "\x89",
    RDISMB      => "\x85",
    ENDEBUG     => "\x86",
    RDYN        => "\x10",
    FLASH_LEN   => 32768,

    BP_CS       => "\x01",
    BP_AUX      => "\x02",
    BP_PULLUP   => "\x04",
    BP_POWER    => "\x08",
};

my %opts;
my $port;
my $time = 500;
my $status_byte;
my $return;
```

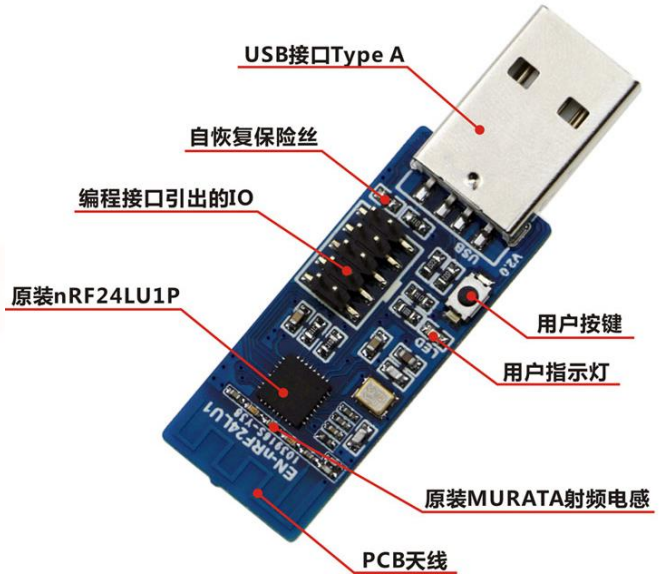
- https://raw.githubusercontent.com/koolatron/buspirate_nrf24lu1p/master/flasher.pl
- The defector standard SPI Flasing script for crazyradio


艰难的更进一步

nRF24LU1+ Based Device

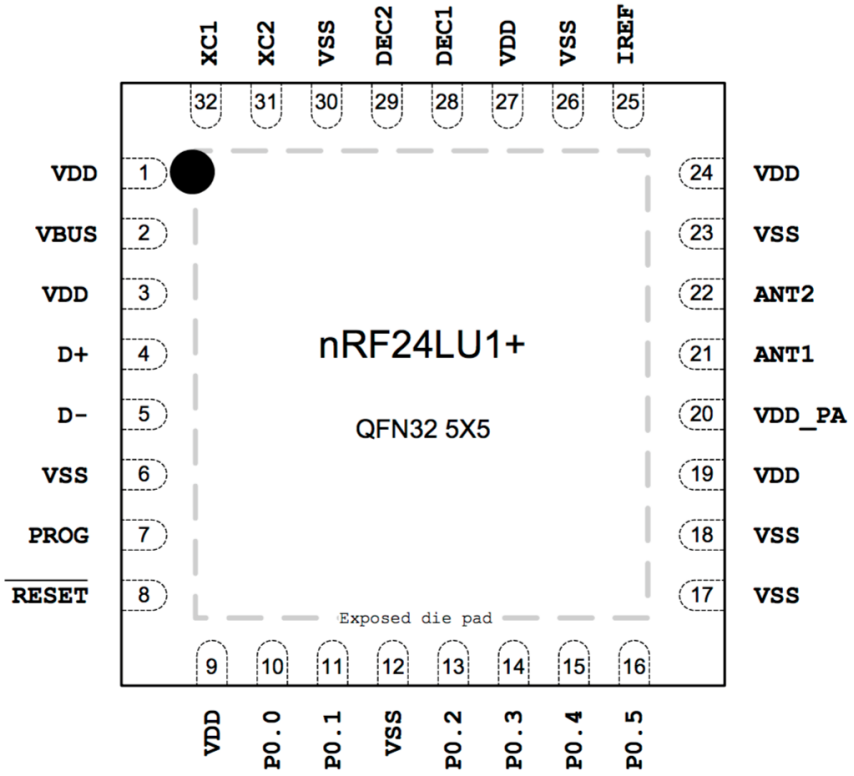






- Comes with nRF24L01 + 2.4 GHz RF transceiver
- USB Connector
- External Antenna or PCB Antenna

nRF24LU1+ Specification



NORDIC SEMICONDUCTOR
Smarter Things

- nRF24LU1 + 2.4 GHz RF transceiver
- Full speed USB 2.0 compliant device controller
- 8-bit microcontroller
- 16 or 32 kilobytes of flash memory
- Up to 12 Mbps air data rate
- Comes with AES encryption acceleration
- Full Spec document in: https://github.com/xwings/tuya

拯救过程

Soldering

Using BusPirate

It's possible to re-program the Crazyradio using a BusPirate and this script via SPI.

Couple of caveats:

- Tested only on OS X. Should work on Linux without modification, and Windows with very minor changes to use the windows serial module.
- It's very slow (~5 minutes to flash the entire .bin file). I deemed this acceptable as this script is for emergency recovery only. I can make it faster if necessary.

Prerequisites:

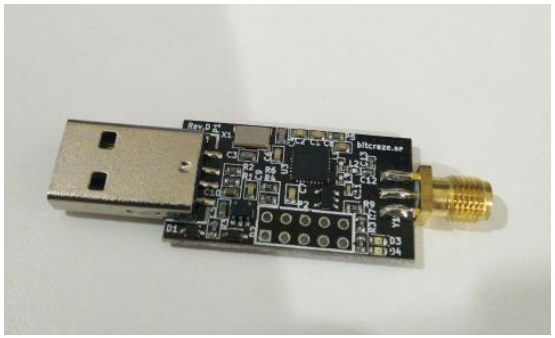
- A Bus Pirate (you should know where to get one of these, if you don't already have one).
- perl and either Device::SerialPort (*nix) or Win32::SerialPort (Windows)
- Some jumper wires to connect the SPI lines on the radio to the ones on the Bus Pirate.

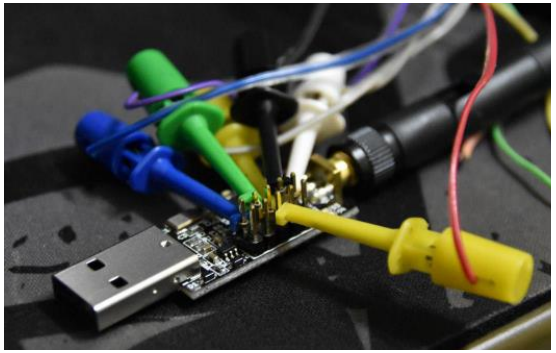
Instructions:

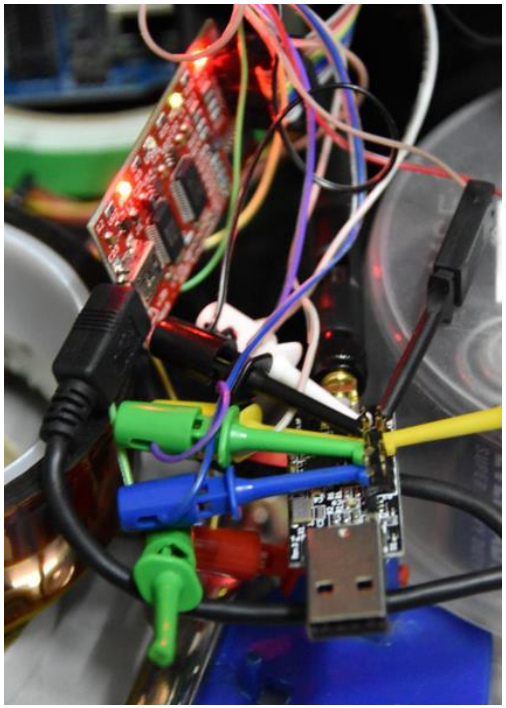
1. Solder a 2×5 pin header onto the programming port of the crazyradio. There's an unpopulated footprint already there for you.
2. Connect the crazyradio to your Bus Pirate using the table below (also noted in the script and readme on git)

```
Bus Pirate       CrazyRadio
============================
MOSI ()      ->  MOSI  (6)
MISO ()      ->  MISO  (8)
SCK  ()      ->  SCK   (4)
CS   ()      ->  CS    (10)
AUX  ()      ->  PROG  (2)
3V3  ()      ->  3V3   (5)
GND  ()      ->  GND   (9)
```

3. Run the script: perl ./flasher.pl -input ./cradio-0.51.bin -device [serial device]
4. Wait till you see lots of hex addresses crawling up your screen. Your device is programming.
5. Go make a sandwich or have a beer (or both).

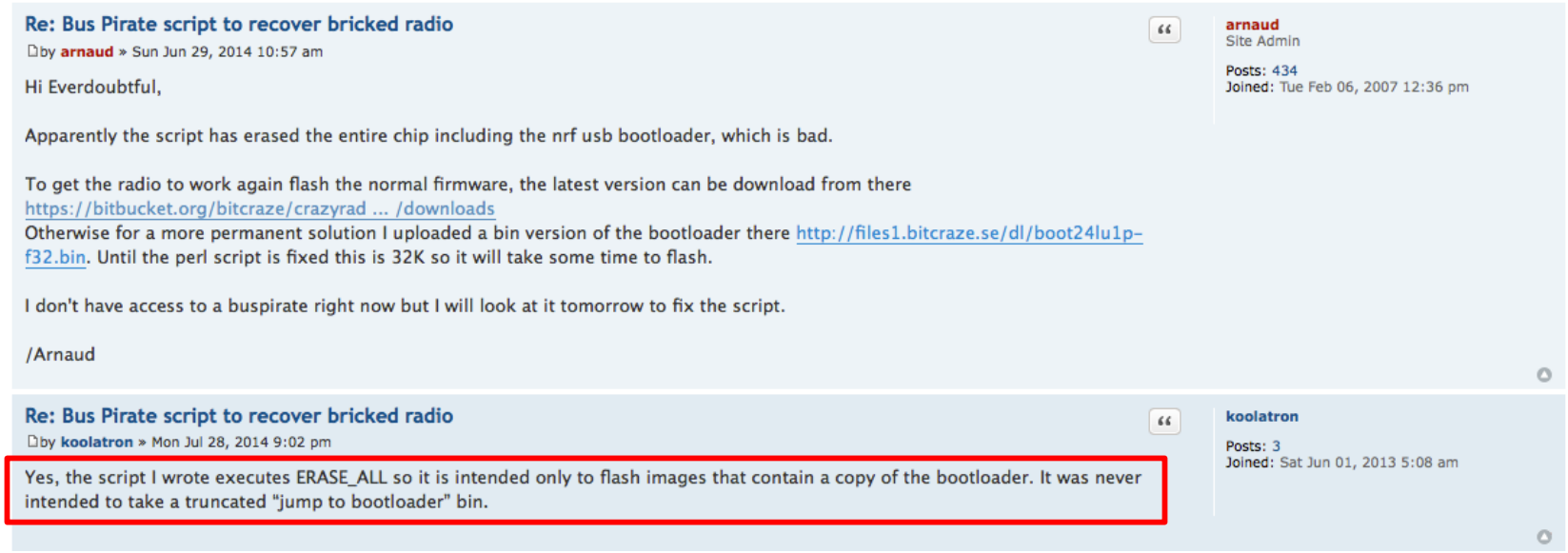






- Crazyradio comes with breakout pin
- Solder a 2 x 5 Pin into crazy radio
- “Clipped” in Bus Pirate accordingly
- Beware of crazyradio breakout pin sequence

Problem 1: Bootloder Missing



Re: Bus Pirate script to recover bricked radio

by **arnaud** » Sun Jun 29, 2014 10:57 am

Hi Everdoubtful,

Apparently the script has erased the entire chip including the nrf usb bootloader, which is bad.

To get the radio to work again flash the normal firmware, the latest version can be download from there
https://bitbucket.org/bitcraze/crazyrad ... /downloads
Otherwise for a more permanent solution I uploaded a bin version of the bootloader there http://files1.bitcraze.se/dl/boot24lu1p-f32.bin. Until the perl script is fixed this is 32K so it will take some time to flash.

I don't have access to a buspirate right now but I will look at it tomorrow to fix the script.

/Arnaud

arnaud
Site Admin

Posts: 434
Joined: Tue Feb 06, 2007 12:36 pm

Re: Bus Pirate script to recover bricked radio

by **koolatron** » Mon Jul 28, 2014 9:02 pm

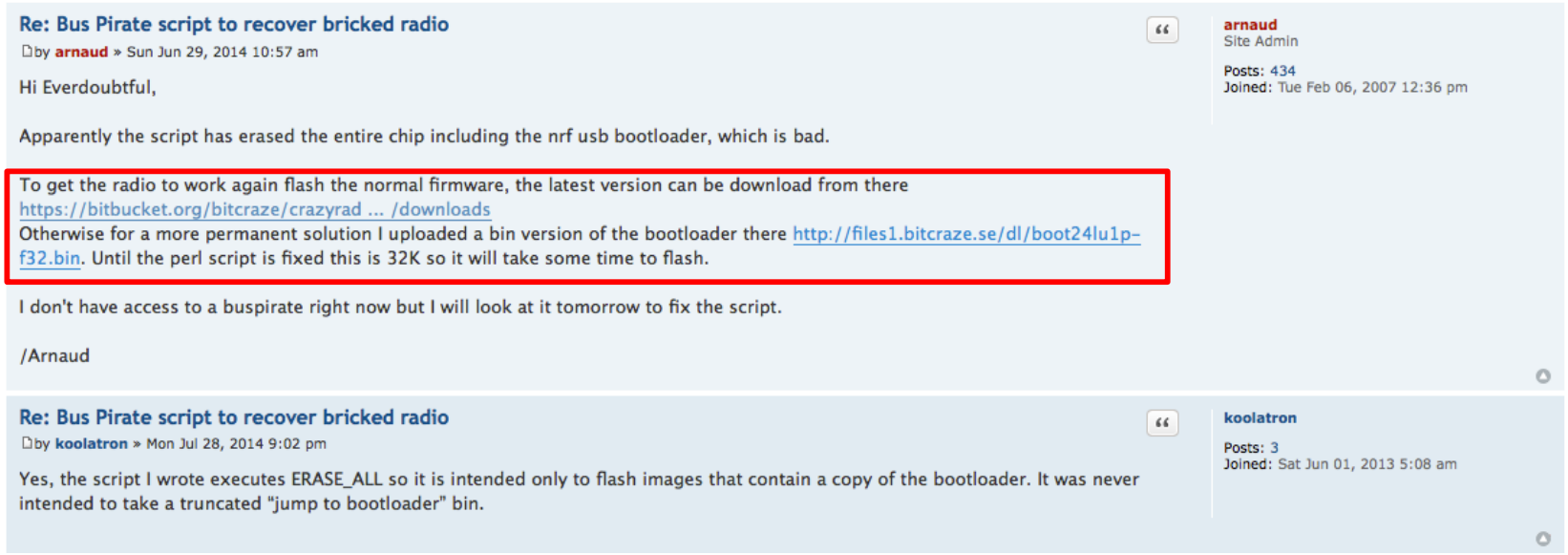
Yes, the script I wrote executes ERASE_ALL so it is intended only to flash images that contain a copy of the bootloader. It was never intended to take a truncated "jump to bootloader" bin.

koolatron

Posts: 3
Joined: Sat Jun 01, 2013 5:08 am

- Due to the “PA” flashed in to the NON “PA”, it overwrites the bootloader
- Almost broken perl script not able to execute completely
- ERASE_ALL makes it all worse
- Info: https://forum.bitcraze.io/viewtopic.php?t=323

The bootloader



Re: Bus Pirate script to recover bricked radio

by **arnaud** » Sun Jun 29, 2014 10:57 am

Hi Everdoubtful,

Apparently the script has erased the entire chip including the nrf usb bootloader, which is bad.

To get the radio to work again flash the normal firmware, the latest version can be download from there
https://bitbucket.org/bitcraze/crazyrad ... /downloads
Otherwise for a more permanent solution I uploaded a bin version of the bootloader there http://files1.bitcraze.se/dl/boot24lu1p-f32.bin. Until the perl script is fixed this is 32K so it will take some time to flash.

I don't have access to a buspirate right now but I will look at it tomorrow to fix the script.

/Arnaud

arnaud
Site Admin

Posts: 434
Joined: Tue Feb 06, 2007 12:36 pm

Re: Bus Pirate script to recover bricked radio

by **koolatron** » Mon Jul 28, 2014 9:02 pm

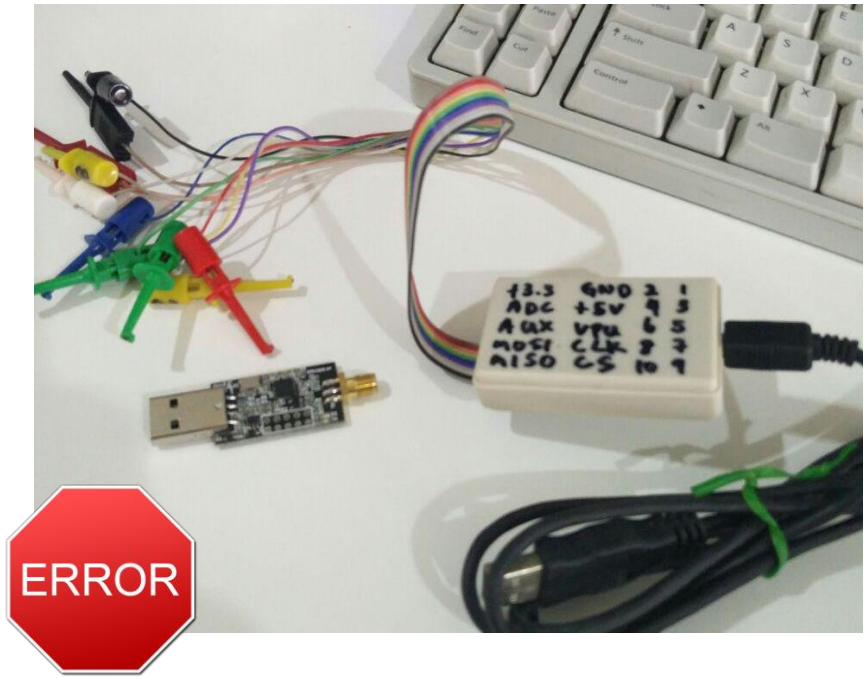
Yes, the script I wrote executes ERASE_ALL so it is intended only to flash images that contain a copy of the bootloader. It was never intended to take a truncated "jump to bootloader" bin.

koolatron

Posts: 3
Joined: Sat Jun 01, 2013 5:08 am

- The possible way is, flash the boot loader
- Once completed, flash the crazyradio firmware
- Bootloder: https://github.com/xwings/tuya

The Final Error



- # git clone https://github.com/RFStorm/mousejack.git
- # cd mousejack
- # make
- Flash the firmware into crazyradio
- Almost working perl script not working

The “Broken” Perl Script

```perl
use strict;
use feature 'say';
use Getopt::Long;
use Device::SerialPort;
use Time::HiRes qw/usleep/;

use constant {
    WREN        => "\x06",
    WRDIS       => "\x04",
    RDSR        => "\x05",
    WRSR        => "\x01",
    READ        => "\x03",
    PROGRAM     => "\x02",
    ERASE_PAGE  => "\x52",
    ERASE_ALL   => "\x62",
    RDFPCR      => "\x89",
    RDISMB      => "\x85",
    ENDEBUG     => "\x86",
    RDYN        => "\x10",
    FLASH_LEN   => 32768,

    BP_CS       => "\x01",
    BP_AUX      => "\x02",
    BP_PULLUP   => "\x04",
    BP_POWER    => "\x08",
};

my %opts;
my $port;
my $time = 500;
my $status_byte;
my $return;

if (!&GetOptions(\%opts,
    'input=s',
    'device=s',
) || ( !$opts{input} && !$opts{device} ) ) {
    die "Please specify both -input <input_file.bin> and -device <Bus Pirate devnode>";
}

$port = new Device::SerialPort( $opts{device} );

# Setup serial

$port->baudrate(115200);
$port->parity("none");
$port->databits(8);
$port->stopbits(1);
$port->buffers(1,1);
$port->write_settings || undef $port;

die "Unable to write settings to serial port." unless $port;

# Setup BP
say "Entering raw bitbang mode...";
    while ( ( $port->read( 5 ) ne "BBIO1" ) && --$time ) {
    $port->write( "\x00" );
    usleep( 20000 );
}
die "Unable to enter raw bitbang mode!" unless $time;
```

```perl
use strict;
use feature 'say';
use Getopt::Long;
use Device::SerialPort;
use Time::HiRes qw/usleep/;

use constant {
    WREN        => "\x06",
    WRDIS       => "\x04",
    RDSR        => "\x05",
    WRSR        => "\x01",
    READ        => "\x03",
    PROGRAM     => "\x02",
    ERASE_PAGE  => "\x52",
    ERASE_ALL   => "\x62",
    RDFPCR      => "\x89",
    RDISMB      => "\x85",
    ENDEBUG     => "\x86",
    RDYN        => "\x10",
    FLASH_LEN   => 32768,

    BP_CS       => "\x01",
    BP_AUX      => "\x02",
    BP_PULLUP   => "\x04",
    BP_POWER    => "\x08",
};

my %opts;
my $port;
my $time = 500;
my $status_byte;
my $return;

if (!&GetOptions(\%opts,
    'input=s',
    'device=s',
) || ( !$opts{input} && !$opts{device} ) ) {
    die "Please specify both -input <input_file.bin> and -device <Bus Pirate devnode>";
}

$port = new Device::SerialPort( $opts{device} );

# Setup serial

$port->baudrate(115200);
$port->parity("none");
$port->databits(8);
$port->stopbits(1);
$port->buffers(1,1);
$port->write_settings || undef $port;

die "Unable to write settings to serial port." unless $port;

# Setup BP
say "Entering raw bitbang mode...";
    while ( ( $port->read( 5 ) ne "BBIO1" ) && --$time ) {
    usleep( 40000 );
}
die "Unable to enter raw bitbang mode!" unless $time;
```

- https://raw.githubusercontent.com/koolatron/buspirate_nrf24lu1p/master/flasher.pl
- Broken by default under VM
- Replace all usleep(20000) to usleep(40000)
- The Fix: https://github.com/xwings/tuya

Re-Flash

bootloader

```
root@kali:~/buspirate_nrf24lu1p# perl flasher.pl -i boot24lu1p-f32.bin -device /
dev/ttyUSB0
Entering raw bitbang mode...
Entering binary SPI mode...
Configuring peripherals...
Configuring SPI...
Enabling programming...
Reading status byte...
Status: 20
Erasing chip...
Enabling programming...
Reading status byte...
Status: 20
Programming device...
0000 : 0278
0002 : 0000
```

firmware

```
root@kali:~/buspirate_nrf24lu1p# perl flasher.pl -input cradio-0.51.bin -device /dev/ttyUSB0
Entering raw bitbang mode...
Entering binary SPI mode...
Configuring peripherals...
Configuring SPI...
Enabling programming...
Reading status byte...
Status: 20
Erasing chip...
Enabling programming...
Reading status byte...
Status: 20
Programming device...
0000 : 0200
0002 : 6b32
0004 : 0000
```

- Two hours for the bootloader
- Two hours for the crazyradio firmware
- Two hours for the mousejack firmware

It Works

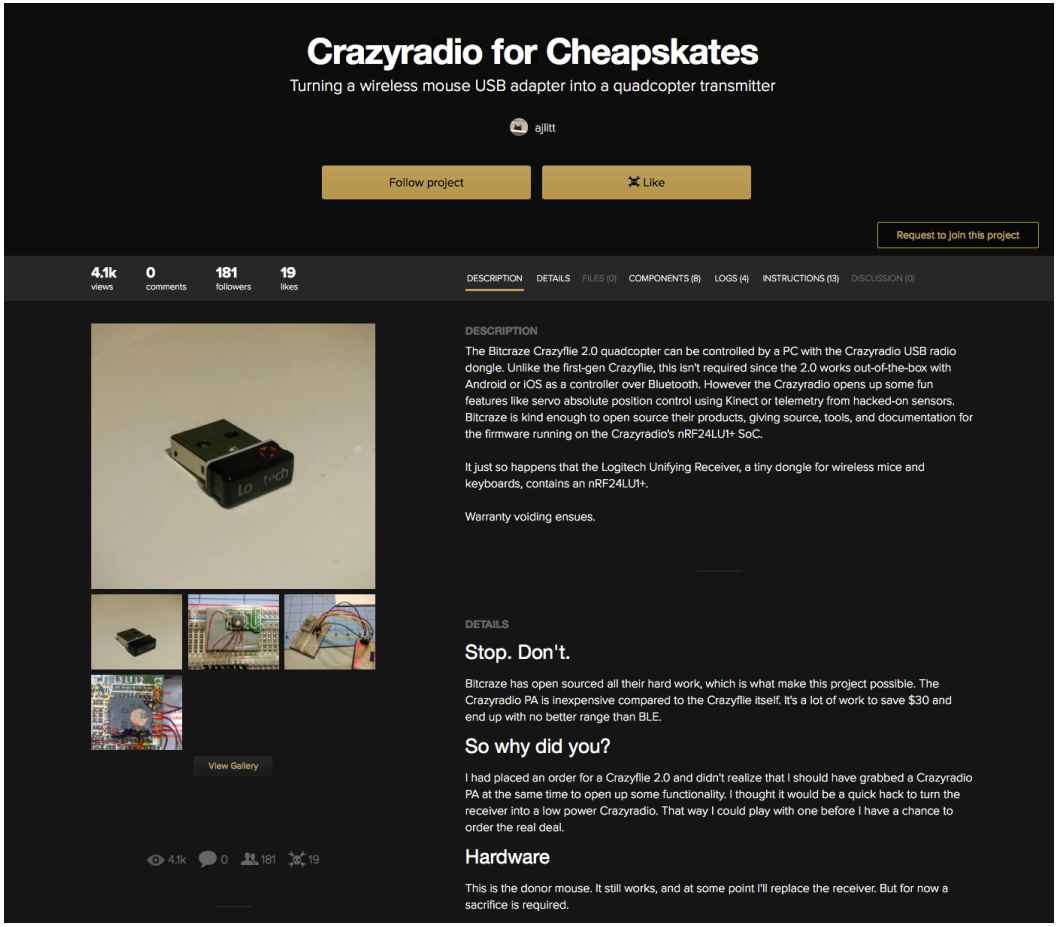
```
[  416.993066] usb 1-2.2: new full-speed USB device number 7 using uhci_hcd
[  417.089596] usb 1-2.2: New USB device found, idVendor=1915, idProduct=0102
[  417.089599] usb 1-2.2: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[  417.089600] usb 1-2.2: Product: Research Firmware
[  417.089601] usb 1-2.2: Manufacturer: RFStorm
```

```
(15)# python ./nrf24-scanner.py
[2016-03-24 21:20:07.388]  32   0  72:E4:
[2016-03-24 21:20:07.425]  32   0  72:E4:
[2016-03-24 21:20:07.458]  32  10  72:E4:                00:C2:00:00:02:D0:FF:00:00:6D
[2016-03-24 21:20:32.988]  32   5  72:E4:                00:40:00:6E:52
```

```
(21)# python ./nrf24-sniffer.py -a 72:E4
[2016-03-24 21:23:08.242]  32   5  72:E4                00:40:00:6E:52
[2016-03-24 21:23:08.335]  32   5  72:E4                00:40:00:6E:52
[2016-03-24 21:23:08.427]  32   5  72:E4                00:40:00:6E:52
[2016-03-24 21:23:08.521]  32  10  72:E4                00:C2:00:00:FA:0F:00:00:00:35
[2016-03-24 21:23:08.529]  32  10  72:E4                00:C2:00:00:F4:0F:00:00:00:3B
[2016-03-24 21:23:08.537]  32  10  72:E4                00:C2:00:00:F0:0F:00:00:00:3F
[2016-03-24 21:23:08.544]  32  10  72:E4                00:C2:00:00:F4:FF:FF:00:00:4C
[2016-03-24 21:23:08.552]  32  10  72:E4                00:C2:00:00:F5:DF:FF:00:00:6B
[2016-03-24 21:23:08.559]  32  10  72:E4                00:C2:00:00:FA:EF:FF:00:00:56
[2016-03-24 21:23:08.569]  32  10  72:E4                00:C2:00:00:FE:FF:FF:00:00:42
[2016-03-24 21:23:08.580]  32  10  72:E4                00:C2:00:00:FE:FF:FF:00:00:42
[2016-03-24 21:23:08.593]  32  10  72:E4                00:4F:00:00:6E:00:00:00:00:43
[2016-03-24 21:23:08.600]  32   5  72:E4                00:40:00:6E:52
[2016-03-24 21:23:08.693]  32   5  72:E4                00:40:00:6E:52
[2016-03-24 21:23:08.732]  32  10  72:E4                00:C2:00:00:00:10:00:00:00:2E
[2016-03-24 21:23:08.739]  32  10  72:E4                00:4F:00:00:6E:00:00:00:00:43
[2016-03-24 21:23:08.756]  32  10  72:E4                00:C2:00:00:01:20:00:00:00:1D
[2016-03-24 21:23:08.763]  32  10  72:E4                00:4F:00:00:6E:00:00:00:00:43
```


结束是另外一个开始

Google Found This



Crazyradio for Cheapskates

Turning a wireless mouse USB adapter into a quadcopter transmitter

ajlitt

Follow project | Like

Request to join this project

4.1k views | 0 comments | 181 followers | 19 likes

DESCRIPTION | DETAILS | FILES (0) | COMPONENTS (8) | LOGS (4) | INSTRUCTIONS (13) | DISCUSSION (0)

View Gallery

4.1k 0 181 19

DESCRIPTION

The Bitcraze Crazyflie 2.0 quadcopter can be controlled by a PC with the Crazyradio USB radio dongle. Unlike the first-gen Crazyflie, this isn't required since the 2.0 works out-of-the-box with Android or iOS as a controller over Bluetooth. However the Crazyradio opens up some fun features like servo absolute position control using Kinect or telemetry from hacked-on sensors. Bitcraze is kind enough to open source their products, giving source, tools, and documentation for the firmware running on the Crazyradio's nRF24LU1+ SoC.

It just so happens that the Logitech Unifying Receiver, a tiny dongle for wireless mice and keyboards, contains an nRF24LU1+.

Warranty voiding ensues.

DETAILS

Stop. Don't.

Bitcraze has open sourced all their hard work, which is what make this project possible. The Crazyradio PA is inexpensive compared to the Crazyflie itself. It's a lot of work to save $30 and end up with no better range than BLE.

So why did you?

I had placed an order for a Crazyflie 2.0 and didn't realize that I should have grabbed a Crazyradio PA at the same time to open up some functionality. I thought it would be a quick hack to turn the receiver into a low power Crazyradio. That way I could play with one before I have a chance to order the real deal.

Hardware

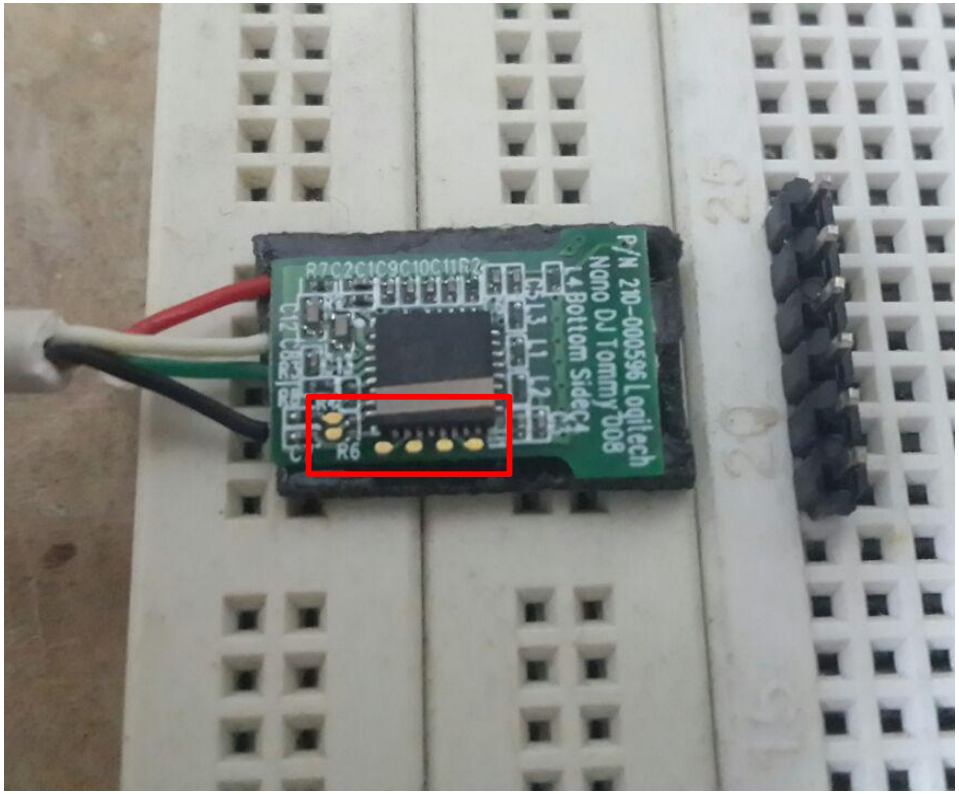
This is the donor mouse. It still works, and at some point I'll replace the receiver. But for now a sacrifice is required.

- We found someone actually trying to fly crazyflies with Logitech unify dongle
- If Logitech Unify dongle compatible with crazyradio firmware, it means ….
- https://hackaday.io/project/6741-crazyradio-for-cheapskates

What is This

- One for all, all for one
- 25 RMB at taobao

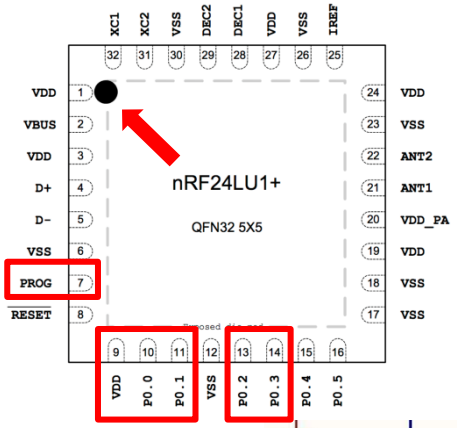
What Is in Logitech Unify Dongle

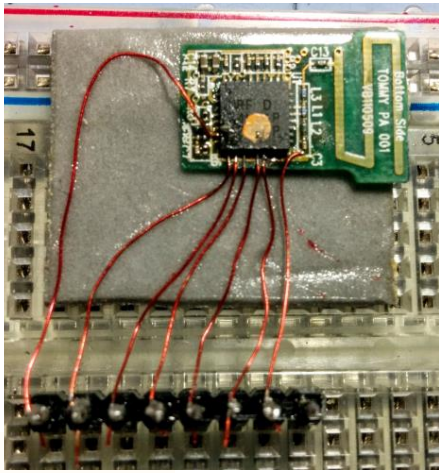


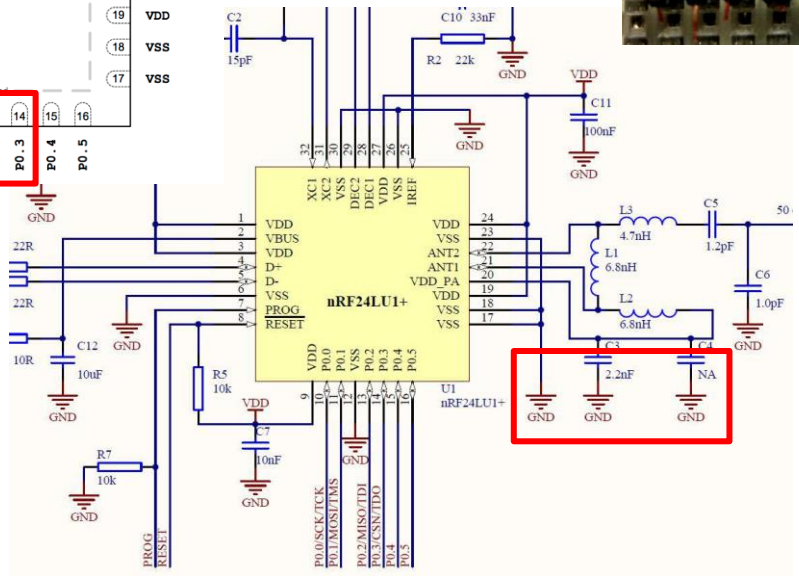
- Open up the casing
- It comes with breakout PINS !
- Find the GRD
- ULTRA STABLE HAND

Identifying the Pins

```
#!/usr/bin/perl -w

# Simple perl script to drive the Bus Pirate and unbrick your CrazyRadio dongle.
# Adapted (sorta) from the Bus Pirate example script and mbed NRF24LU1+ flasher projects:
#  http://code.google.com/p/the-bus-pirate/source/browse/trunk/scripts/SPIeeprom.pl
#  http://mbed.org/users/mux/code/nrflash
#
# This script uses the aux output on the Bus Pirate as the PROG pin on the CrazyRadio's NRF24LU1+ chip.
#
# Electrical connections are as follows:
#
#  Bus Pirate       CrazyRadio
#  ==========================
#  MOSI ()      ->  MOSI   (6)
#  MISO ()      ->  MISO   (8)
#  SCK  ()      ->  SCK    (4)
#  CS   ()      ->  CS     (10)
#  AUX  ()      ->  PROG   (2)
#  3V3  ()      ->  3V3    (5)
#  GND  ()      ->  GND    (9)

use strict;
use feature 'say';
use Getopt::Long;
use Device::SerialPort;
use Time::HiRes qw/usleep/;

use constant {
    WREN        => "\x06",
    WRDIS       => "\x04",
    RDSR        => "\x05",
    WRSR        => "\x01",
    READ        => "\x03",
    PROGRAM     => "\x02",
    ERASE_PAGE  => "\x52",
    ERASE_ALL   => "\x62",
    RDFPCR      => "\x89",
    RDISMB      => "\x85",
    ENDEBUG     => "\x86",
    RDYN        => "\x10",
    FLASH_LEN   => 32768,

    BP_CS       => "\x01",
    BP_AUX      => "\x02",
    BP_PULLUP   => "\x04",
    BP_POWER    => "\x08",
};

my %opts;
my $port;
my $time = 500;
my $status_byte;
my $return;
```

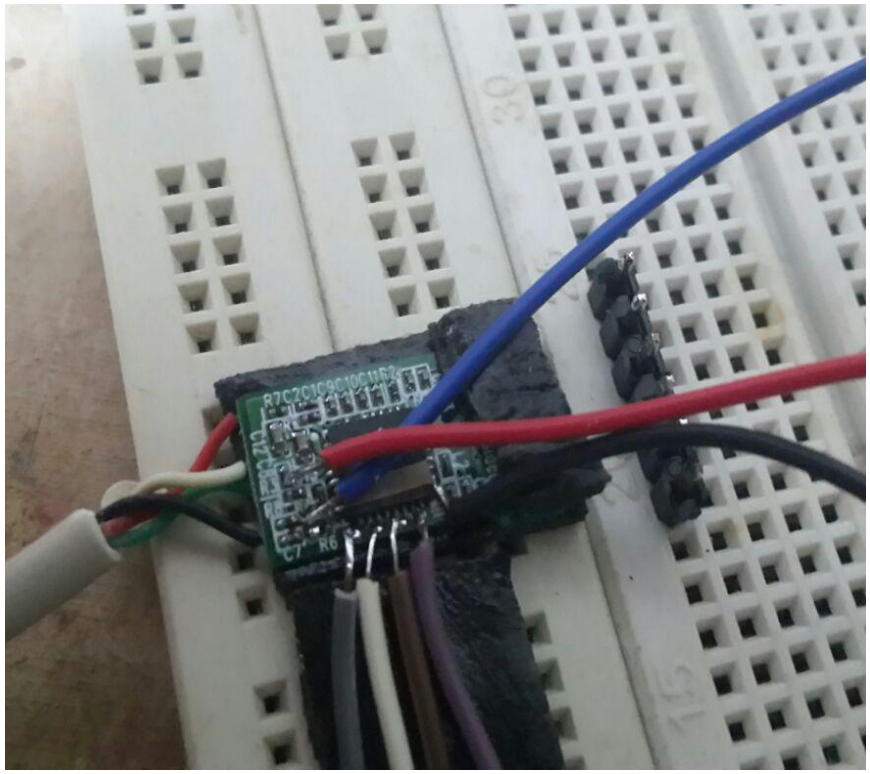


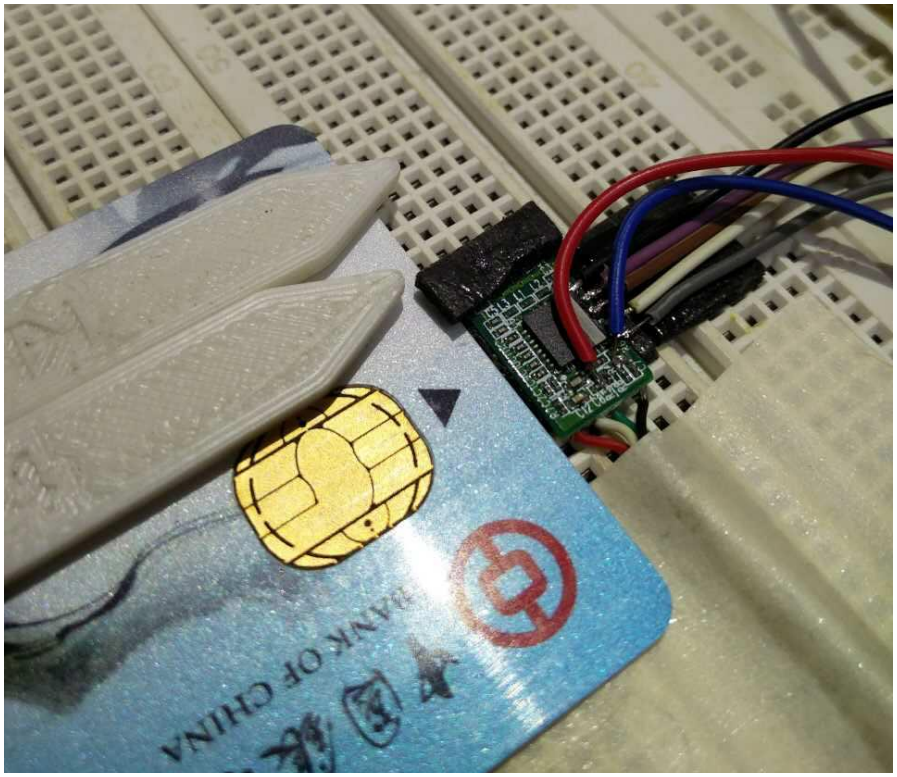




- MOSI - Pin 11
- MISO - Pin 13
- SCK - Pin 10
- CS - PIN 14
- AUX – Pin 7
- 3V3 - Pin 1
- GND – Any GND

Soldering





- Breakout Pin save the world
- Soldering all the Pin accordingly
- Connects to BUS Pirate
- Start Flashing the boot loader
- Flash MouseJack firmware

Re-Flash

bootloader

```
root@kali:~/buspirate_nrf24lu1p# perl flasher.pl -i boot24lu1p-f32.bin -device /
dev/ttyUSB0
Entering raw bitbang mode...
Entering binary SPI mode...
Configuring peripherals...
Configuring SPI...
Enabling programming...
Reading status byte...
Status: 20
Erasing chip...
Enabling programming...
Reading status byte...
Status: 20
Programming device...
0000 : 0278
0002 : 0000
```

firmware

```
root@kali:~/buspirate_nrf24lu1p# perl flasher.pl -input cradio-0.51.bin -device /dev/ttyUSB0
Entering raw bitbang mode...
Entering binary SPI mode...
Configuring peripherals...
Configuring SPI...
Enabling programming...
Reading status byte...
Status: 20
Erasing chip...
Enabling programming...
Reading status byte...
Status: 20
Programming device...
0000 : 0200
0002 : 6b32
0004 : 0000
```

- Two hours for the bootloader
- Two hours for the crazyradio firmware
- Two hours for the mousejack firmware


One More Thing

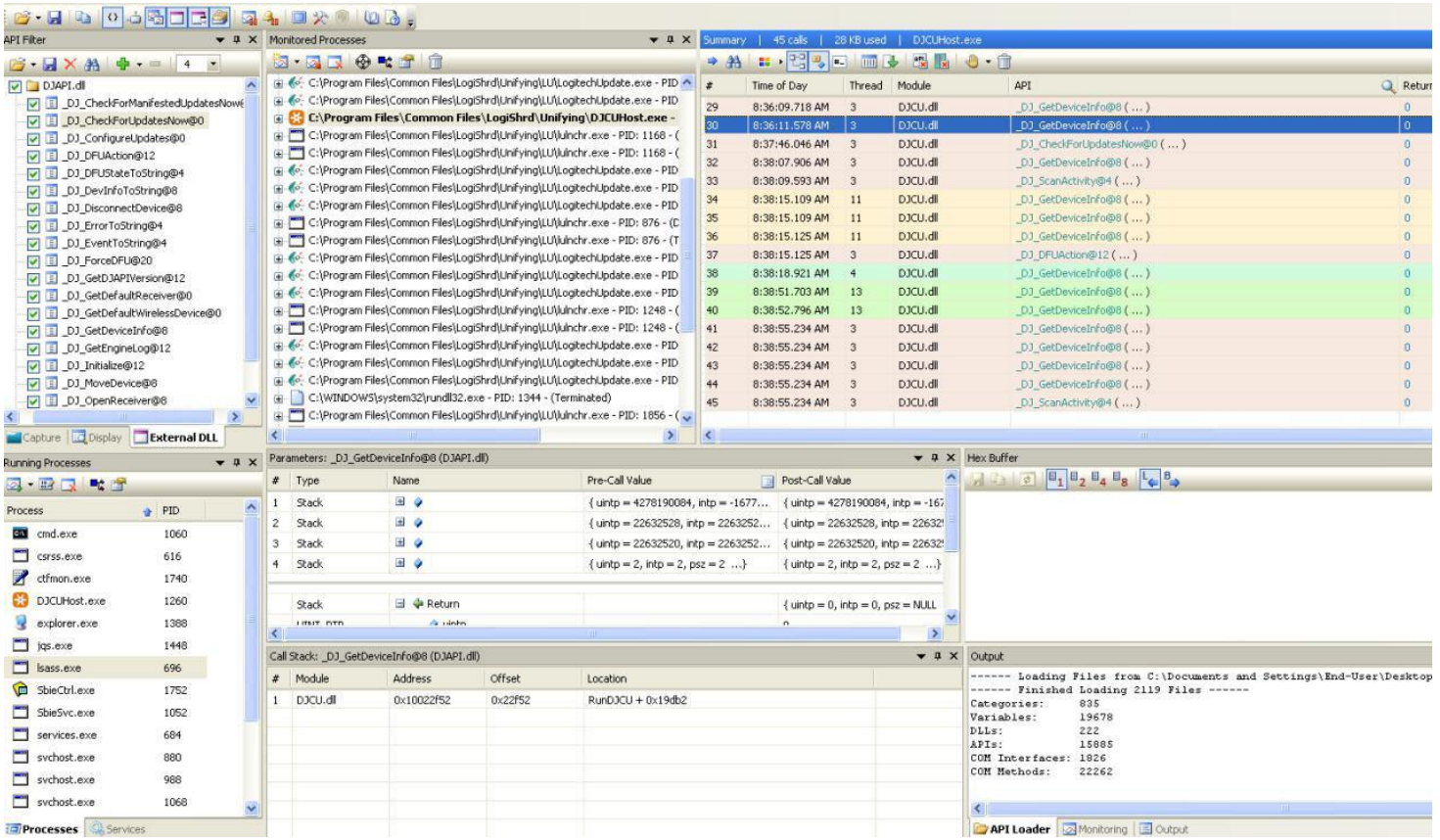
What Is Missing

- Is there really why it call MouseJack?
- Only Mouse at the moment?
- Possible to hijack a keyboard?

Having Fun with Logitech Keyboard

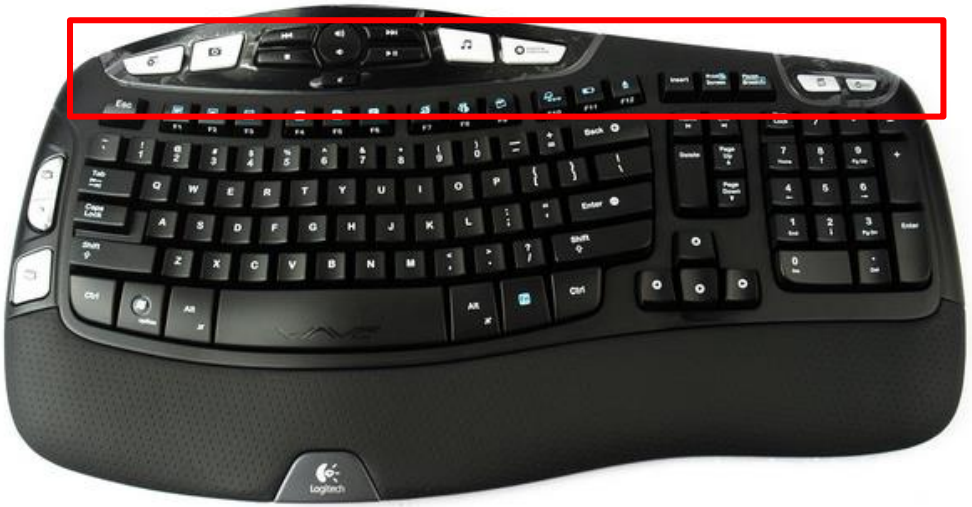
- Most popular brand, Logitech
- Lets see what is in Logitech Keyboard

What We Know



- AES 128bit encryption between keyboard and dongle
- Able to dump some functions
- Few projects doing Logitech Unifying Keyboard, such as solaar. http://pwr.github.io/Solaar
- Time is too limited and nothing much able to capture from the trace

Some Info on Wireless Keyboard



- Most of the multimedia key seems to be not being encrypted
- Not enough to encrypt all the keys ?

What If, Keyboard Is Not Available

- No one will bring a wireless keyboard outdoor
- Send in unencrypted keystroke to mouse dongle? Yes, it works
- Sending encrypted keystroke using unencrypted method. Example, brute force?
- Or Presenter ?

Dumping the Firmware

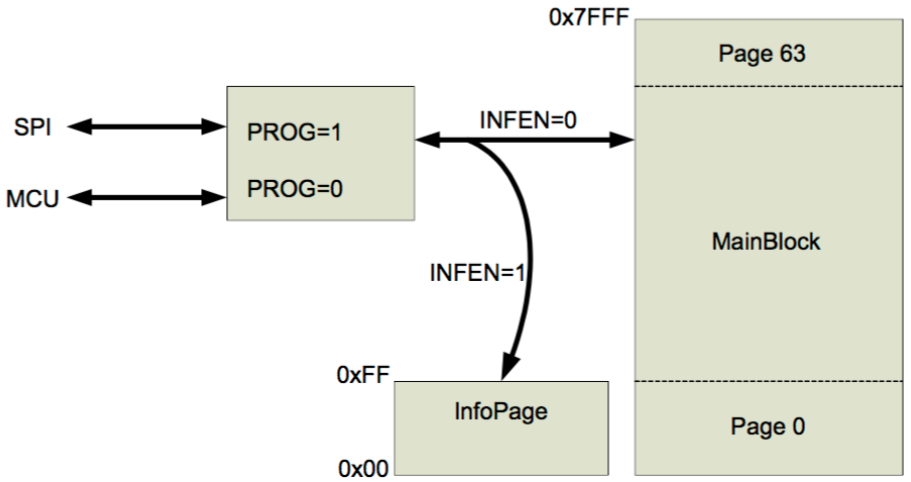


Figure 62. Flash memory block diagram

- Almost Not Possible
- InfoPage Readback blocking, 0x22
- MainBlock Readback blocking, 0x23
- Only can perform complete rewrite

Getting the Firmware

Reply | Topic Options | Message Listing | Previous Topic | Next Topic

ModeratorTeam
Moderator
Posts: 227
Registered: 08-25-2010

Logitech Response to Unifying Receiver Research Findings [Edited]

Options

02-23-2016 09:10 AM - edited 02-24-2016 03:16 PM

You may have read or heard that researchers from Bastille Security found a potential vulnerability in Logitech's Unifying receiver. The Unifying receiver allows you to connect multiple compatible keyboards and mice to a laptop or desktop computer with a single USB receiver.

Bastille Security approached us regarding their work. We have been in regular communication with them since and together have discussed their findings.

Bastille Security identified the vulnerability in a controlled, experimental environment. The vulnerability would be complex to replicate and would require physical proximity to the target. It is therefore a difficult and unlikely path of attack.

We have nonetheless taken Bastille Security's work seriously and developed a firmware fix. If you have concerns, and would like to ensure this vulnerability is eliminated, you can follow these steps:

1. Download and install **Unifying Software**.
2. To check the firmware version on your Unifying receiver, go to **Unifying Software → Advanced**, and then select the **Unifying Receiver**.

3. The version of the firmware is listed in the right pane.
   - If the firmware version is in 0**12**.xxx.000xx format, download and save RQR_012_005_000**28**.exe through the following link: **http://logt.ly/0222**
   - If the firmware version is in 0**24**.xxx.000xx format, download and save RQR_024_003_000**27**.exe through the following link: **http://logt.ly/0224**
4. Run the downloaded firmware package.
5. Open **Unifying Software → Advanced**, then select the **Unifying Receiver.**
6. In the right pane, click on **Update Firmware** and wait until the firmware update is complete.

Note: To have all the features working correctly after updating the firmware, please ensure that you have the latest version of **SetPoint** and/or **Options** software that supports your device.

Logitech's Unifying technology was launched in 2007 and has been used by millions of our consumers since. To our knowledge, we have never been contacted by any customer with such an issue related to this potential vulnerability.

- Download
- Simple RE
- Got the firmware in HEX
- https://github.com/xwings/tuya

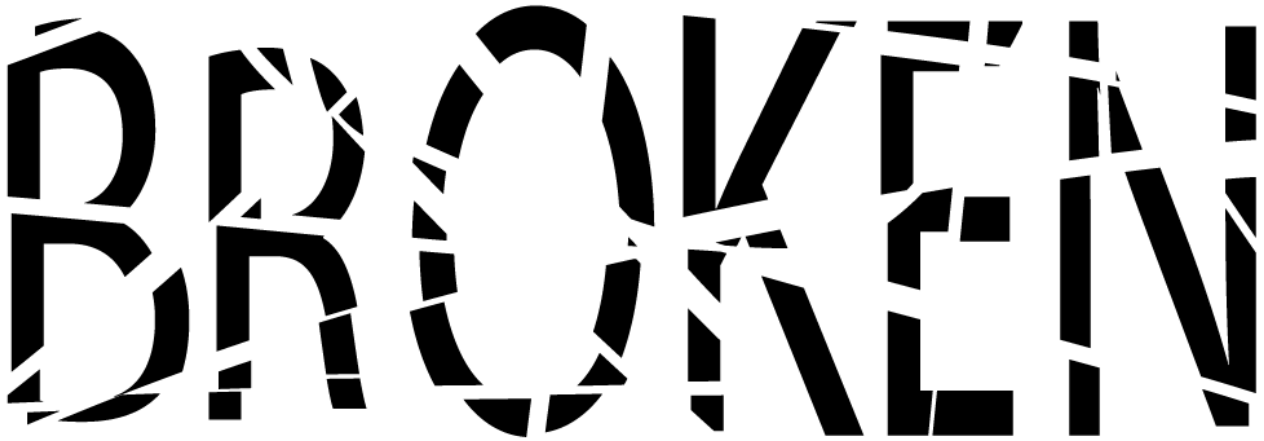
Intel 8051

```
code:0000145D E5 3A
code:0000145F 70 12
code:00001461 75 76 05
code:00001464 75 77 01
code:00001467 90 83 7E
code:0000146A E0
code:0000146B 54 FE
code:0000146D F0
code:0000146E 54 FD
code:00001470 F0
code:00001471 01 30
code:00001473
code:00001473
code:00001473
code:00001473 E5 3A
code:00001475 B4 01 0F
code:00001478 75 76 05
code:0000147B 75 77 02
code:0000147E 90 83 7E
code:00001481 E0
code:00001482 54 FD
code:00001484 F0
code:00001485 21 38
code:00001487
code:00001487
code:00001487
code:00001487 E5 3A
code:00001489 B4 02 0F
code:0000148C 75 76 05
code:0000148F 75 77 03
code:00001492 90 83 7E
code:00001495 E0
code:00001496 44 02
code:00001498 F0
code:00001499 21 86
code:0000149B
code:0000149B
code:0000149B
code:0000149B E5 3A
code:0000149D 64 03
code:0000149F 70 58
code:000014A1 F5 76
```

```
                mov     A, 0x3A
                jnz     code_1473
                mov     0x76, #5
                mov     0x77, #1
                mov     DPTR, #0x837E
                movx    A, @DPTR
                anl     A, #0xFE
                movx    @DPTR, A
                anl     A, #0xFD
                movx    @DPTR, A
                ajmp    code_1030
; ---------------------------------------------------------------------------

code_1473:                              ; CODE XREF: code_143A+25↑j
                mov     A, 0x3A
                cjne    A, #1, code_1487
                mov     0x76, #5
                mov     0x77, #2
                mov     DPTR, #0x837E
                movx    A, @DPTR
                anl     A, #0xFD
                movx    @DPTR, A
                ajmp    code_1138
; ---------------------------------------------------------------------------

code_1487:                              ; CODE XREF: code_143A+3B↑j
                mov     A, 0x3A
                cjne    A, #2, code_149B
                mov     0x76, #5
                mov     0x77, #3
                mov     DPTR, #0x837E
                movx    A, @DPTR
                orl     A, #2
                movx    @DPTR, A
                ajmp    code_1186
; ---------------------------------------------------------------------------

code_149B:                              ; CODE XREF: code_143A+4F↑j
                mov     A, 0x3A
                xrl     A, #3
                jnz     code_14F9
                mov     0x76, A
```

- Convert the HEX to BIN
- 32k file for nRF24LU1
- Hunt for the encryption lib call
- Question, What is the key or where is the key
- Learn Intel 8051 Assembly

Why If It Really Breakable



- What if MouseJack team actually breaks the keyboard encryption
- Broken, will be broken forever
- It is possible to break the encryption. Why?

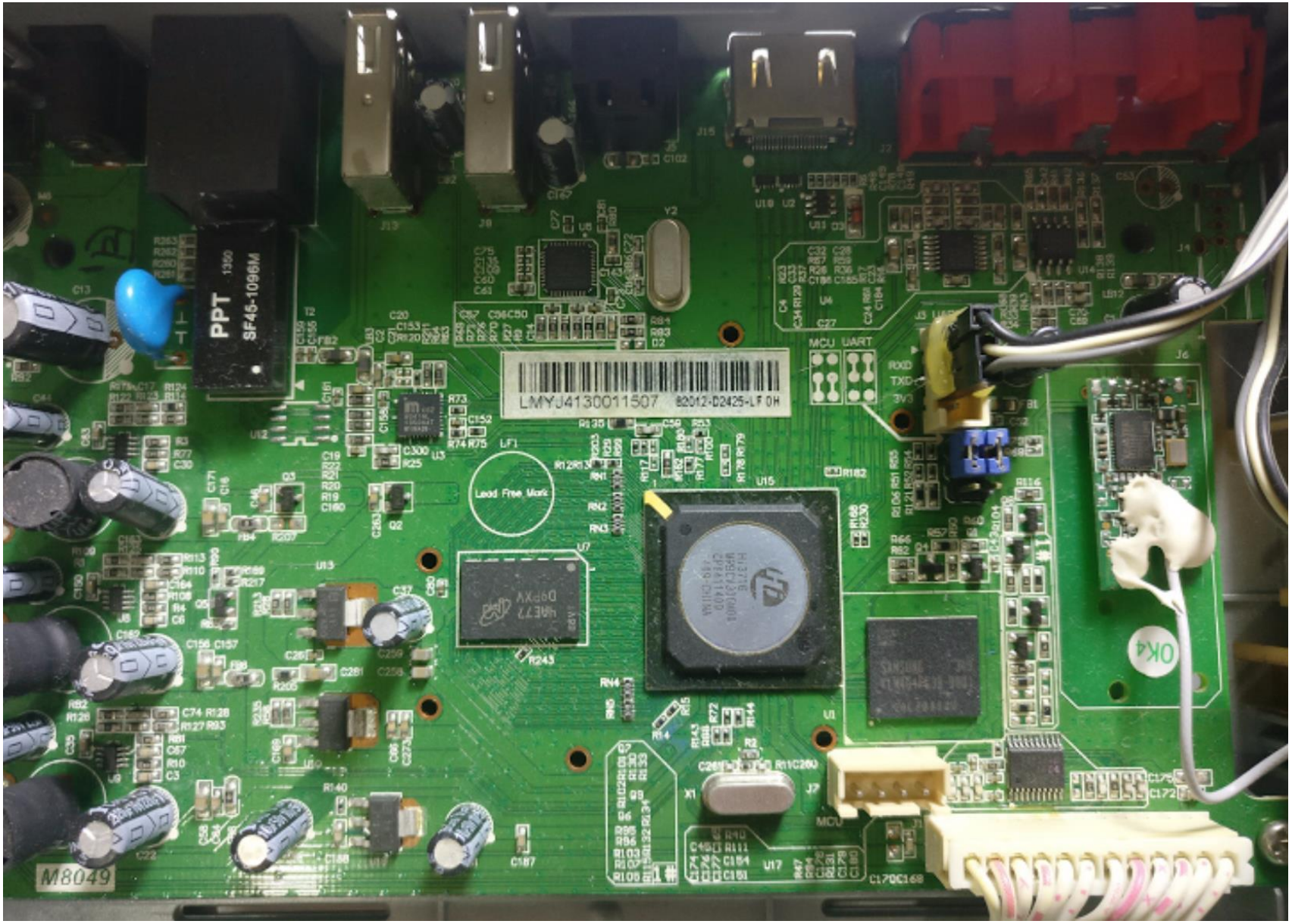

午休


复习时间

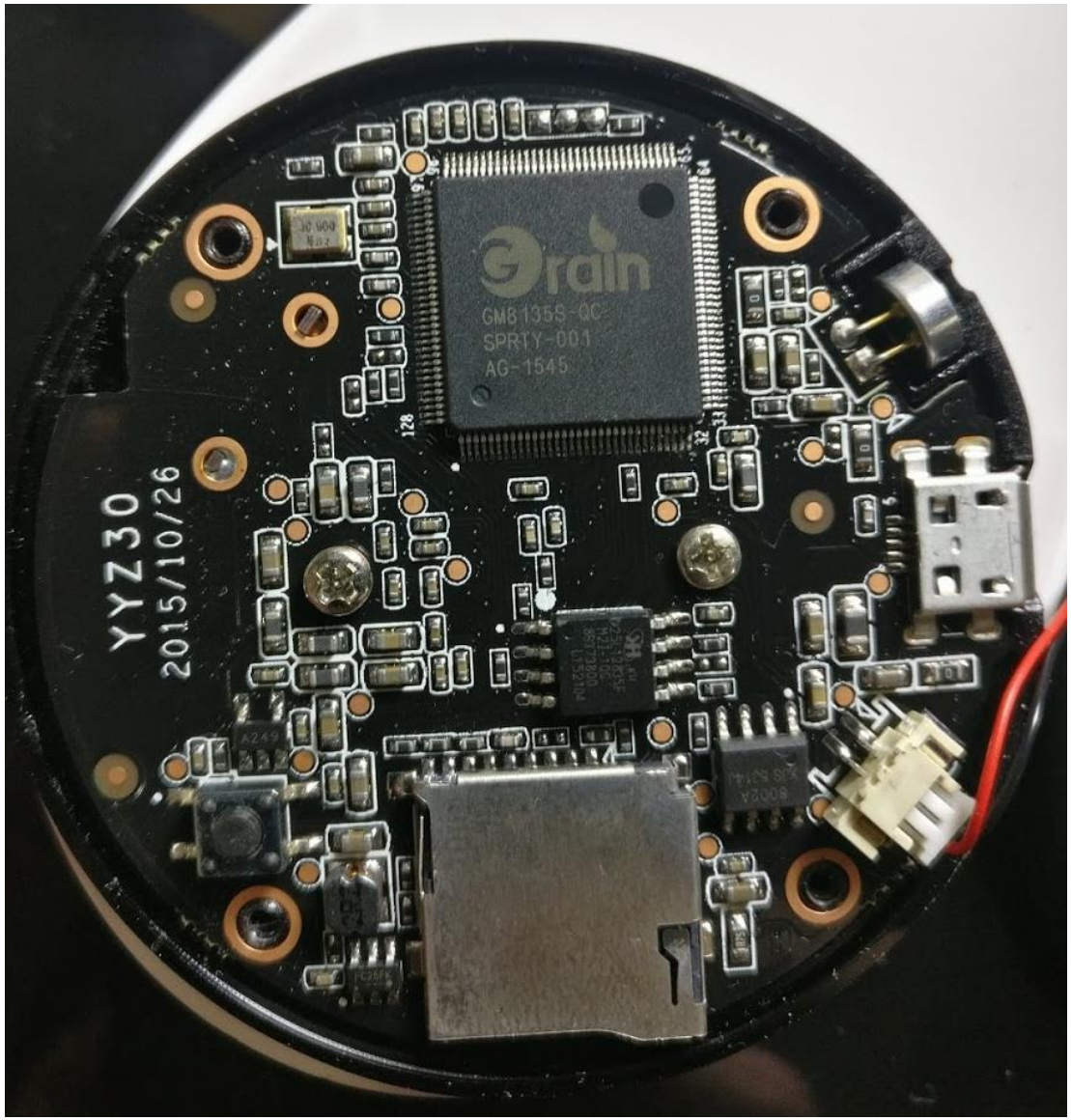
会变动的UART


IR9410
TOP: M
161214
S1604303590
BS1003067C031978 1MY TV IR-9410
DIP/SMD
IR9410-REV1.0

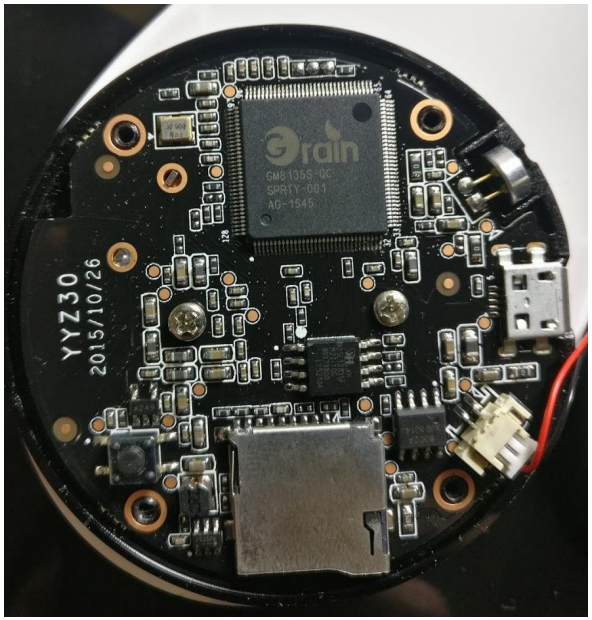
会变动的UART


PPT
SF45-1096M
LMYJ4130011507
MCU UART
RXD
TXD
3V3
Lead Free Work
M8049

真的没有吗 - 0



真的没有吗



2.2 Pin Assignments

Figure 2-1through Figure 2-4 show the pin assignments of GM8136S/GM8135S.

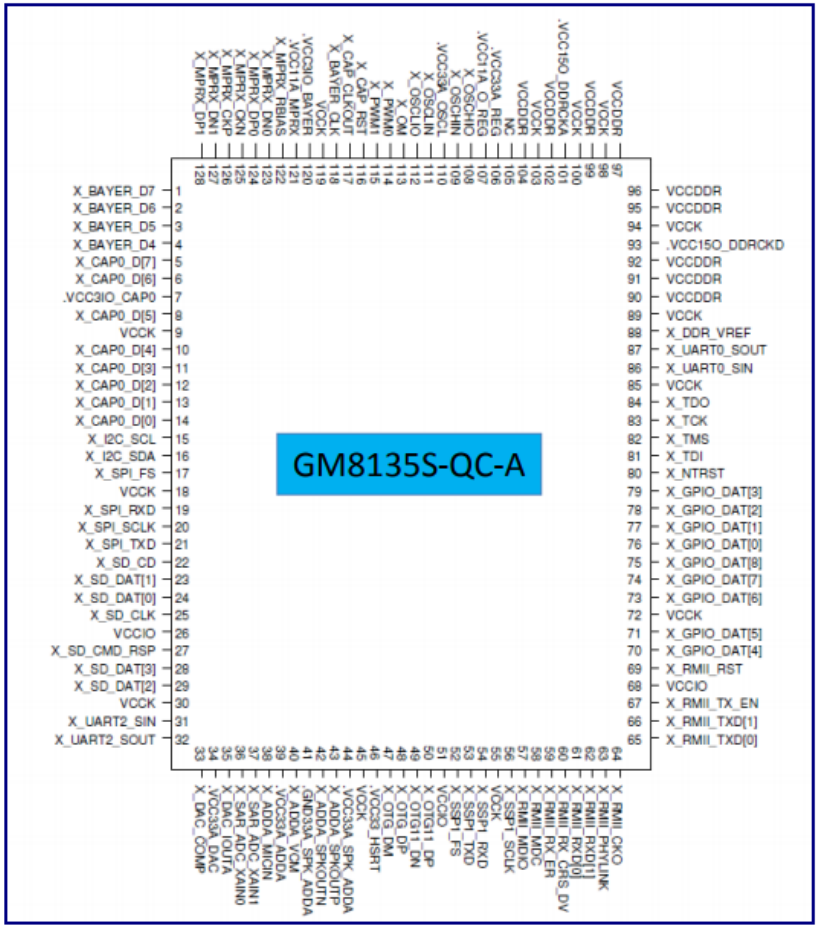
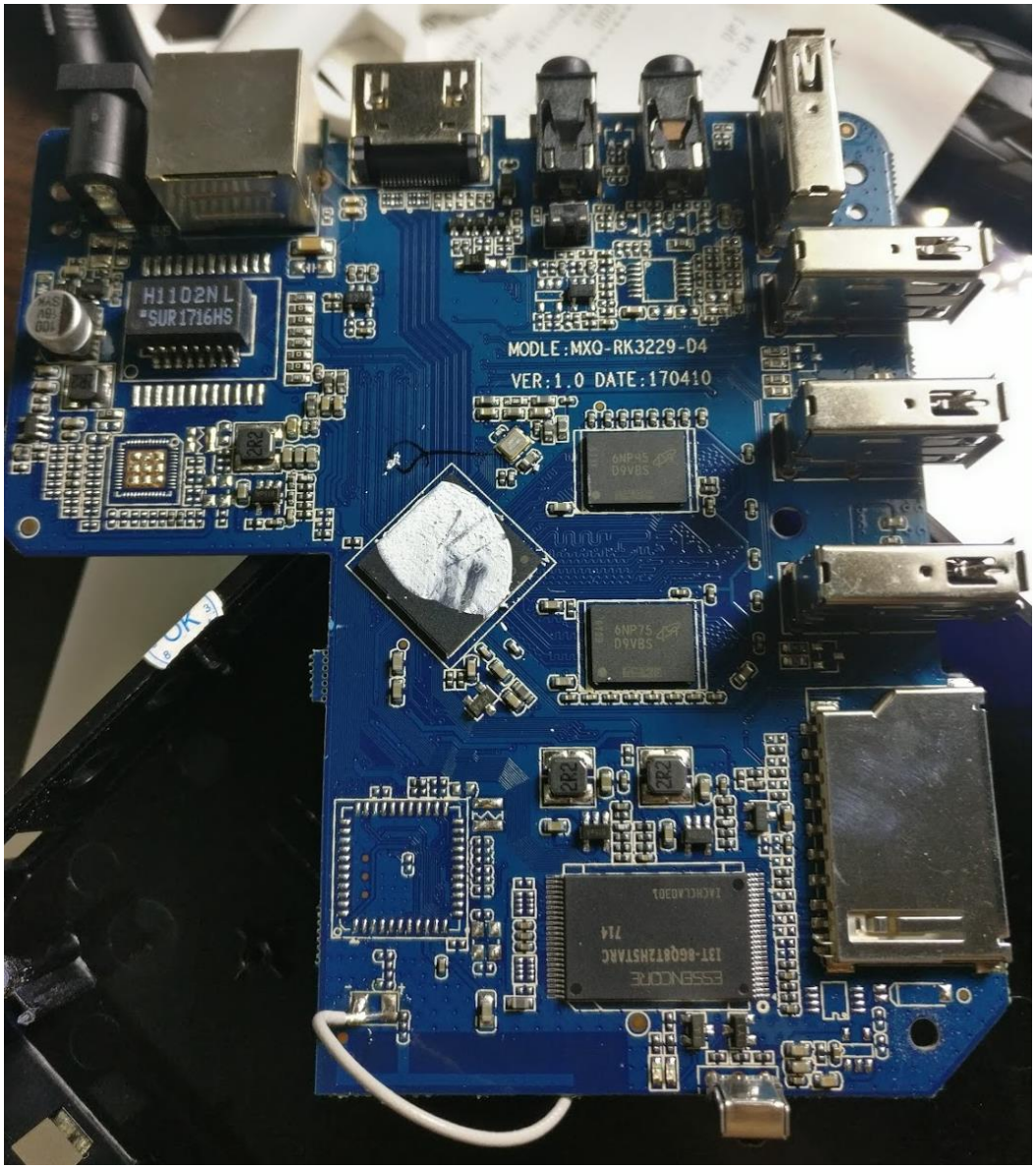


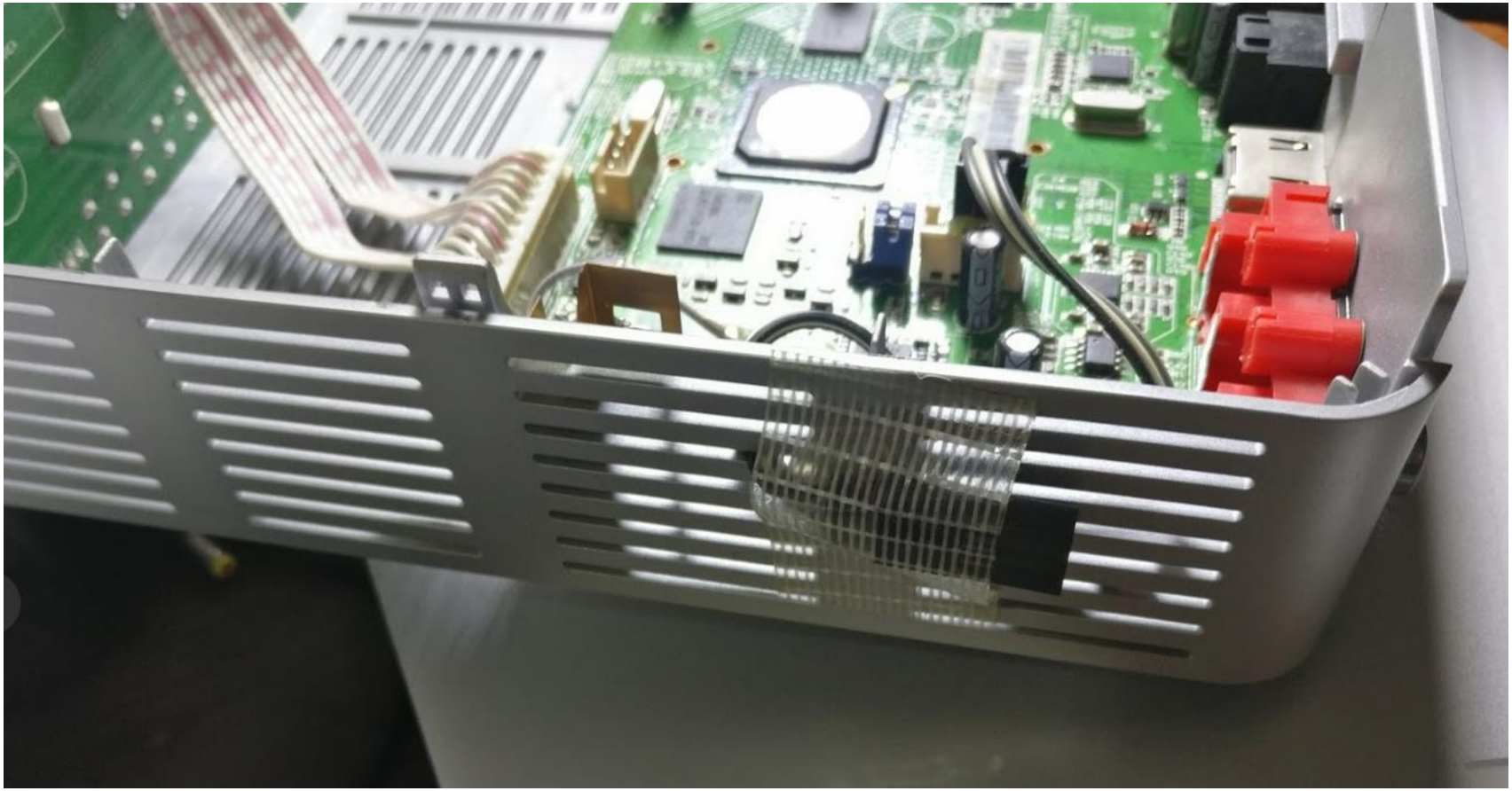
Figure 2-1. Pin Assignments of LQFP (LQFP128) of GM8135S-QC-A (Top View)

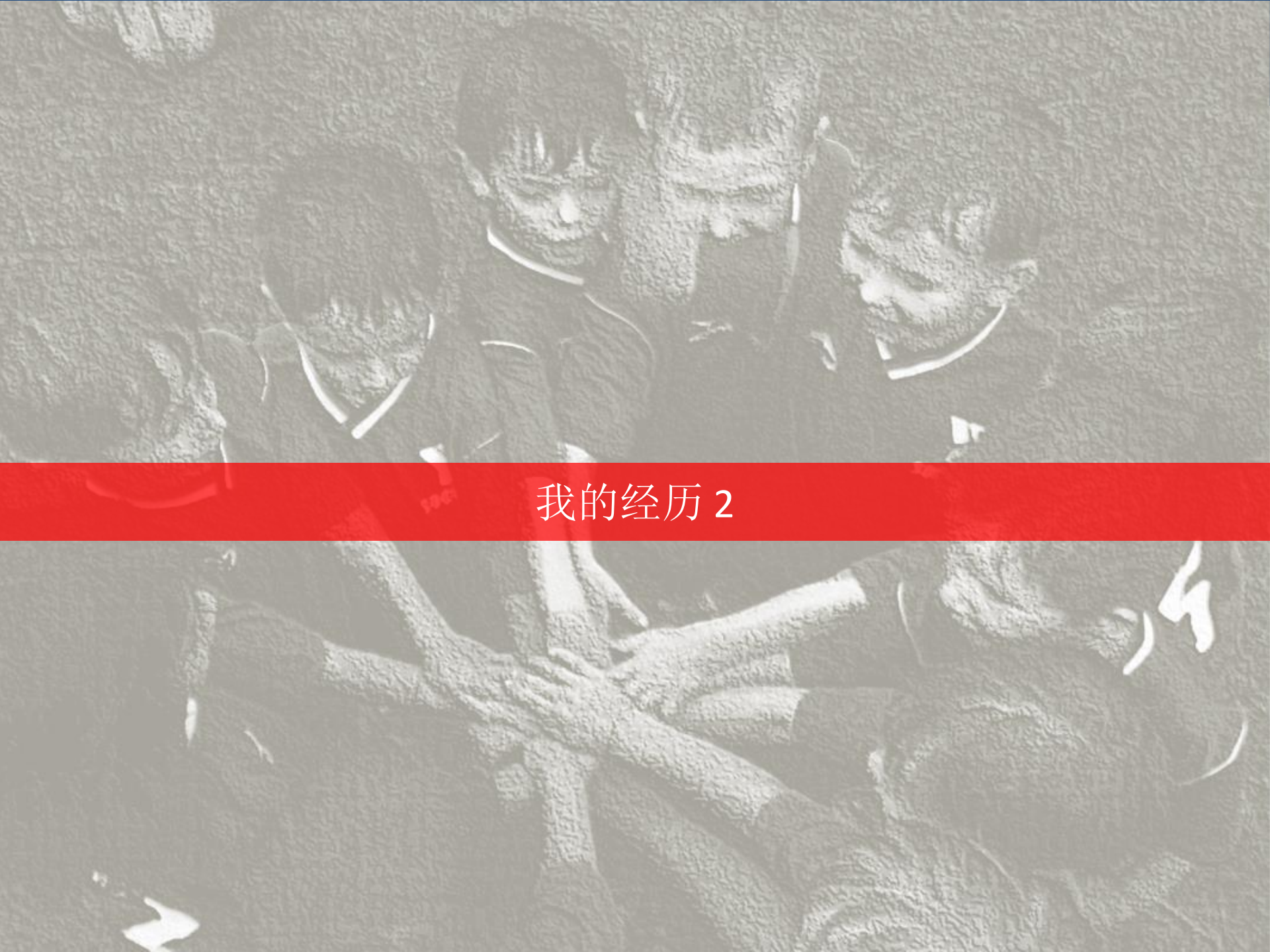
真的没有吗 - 1



提示

给自己留后门





我的经历 2

网购-马来西亚人的悲哀





分享　收藏商品（39319人气）　举报

小蚁智能摄像机1080p一代升级版高清夜视手机网络监控摄像头无线

菜鸟发货 只换不修

天猫电器城 正快省新　闪电到家 超值包邮

全球3C家电狂欢周　此商品8月14日开卖，请提前加入购物车

天猫 购物券　全天猫实物商品通用　去刮券

专柜价　~~¥169.01-219.01~~

价格　**¥169.00-219.00**

运费　浙江嘉兴 至 杭州 上城区 清波街道 快递 0.00

次日达·菜鸟联盟　24:00前付款，承诺8月13日送达

月销量 **8567**　累计评价 **25315**　送天猫积分 **16**起

颜色分类

1080p智能摄像机一代升级版

1080p智能摄像机一代升级版+16G内存卡

1080p智能摄像机一代升级版+30天云存储充值卡

数量　1　件　库存49件

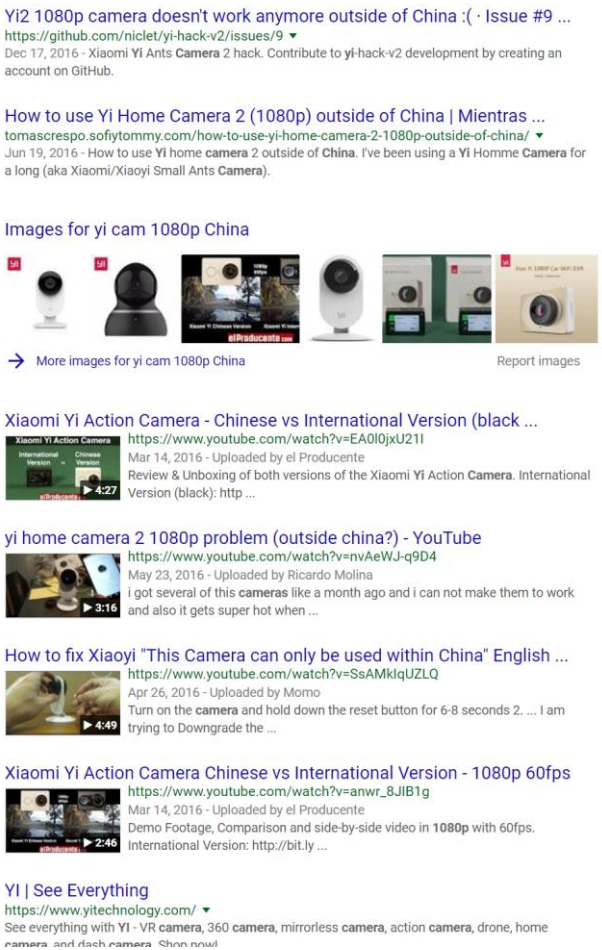
立即购买　加入购物车

服务承诺　超值包邮　闪电到家　正品保证　只换不修　支付方式

极速退款　赠运费险　七天无理由退换

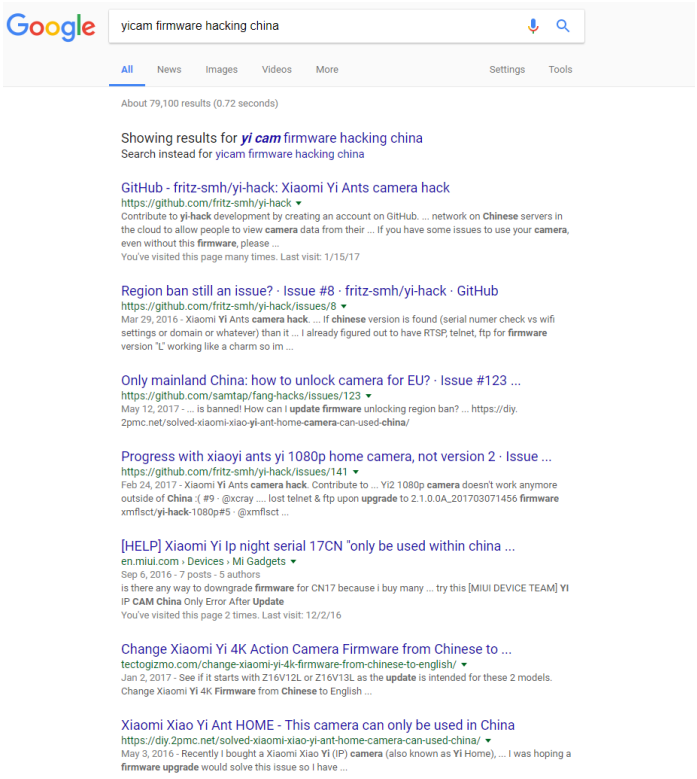
开箱

会说话的摄像头



- 我是马来西亚人
- 我住在马来西亚
- 马来西亚没有卖马来西亚版
- 我在淘宝买了中国版
- 我被警告这是不能在国外用的

百(google)度(.com)



- 国内和国外答案

一个伤心的旅程的开始

[SOLVED] Xiaomi Xiao Yi Ant HOME — This camera can only be used in China (1.8.6.1)

In IT DIY Tags firmware, hack, pentesting May 3, 2016 Csaba Peter

Recently I bought a Xiaomi Xiao Yi (IP) camera (also known as Yi Home), Chinese version. The camera looks nice, the picture quality is ok, and worked fine on my local Wifi.

However, I was unfortunate enough to receive and test the camera when Xiaomi decided to deny access from the iOS app to the camera outside of China (error 5400). I was hoping a firmware upgrade would solve this issue so I have upgraded from 1.8.5.1L to 1.8.6.1B. Now my camera was useless. The camera would say "This camera can only be used in China" and would shut down.

This was the tipping point when I have decided I will investigate what's happening with this camera and what can be done to make it functional again. At the time of writing the remote access (error 5400) has been solved by the provider so no additional action is required. (I tried to convert a CN camera to international one by changing the serial of the device, but couldn't test from a European or US IP and probably I would have needed access to the system files of a functional international camera to compare)

So the remaining issue was the camera shut down with the latest firmware (tested with 1.8.6.1A and 1.8.6.1B).

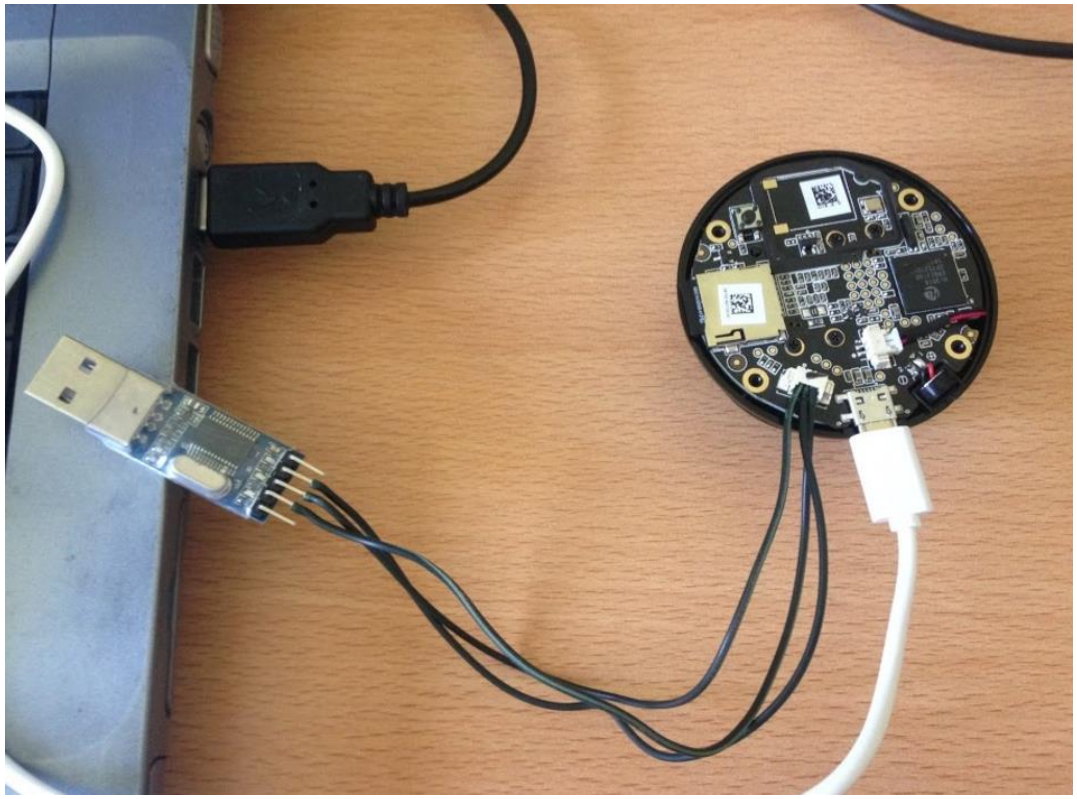
If you do a search there are heaps of websites describing how you can gain access to the camera and ultimately enable remote access via telnet. I won't get into those details, you can check some of the websites I listed below.

Once you logged into the camera via telnet the fun part begins. The camera is running a Linux version.

```
# uname -a
Linux (none) 3.0.8 #1 Wed Apr 30 16:56:49 CST 2014 armv5tejl GNU/Linux
```

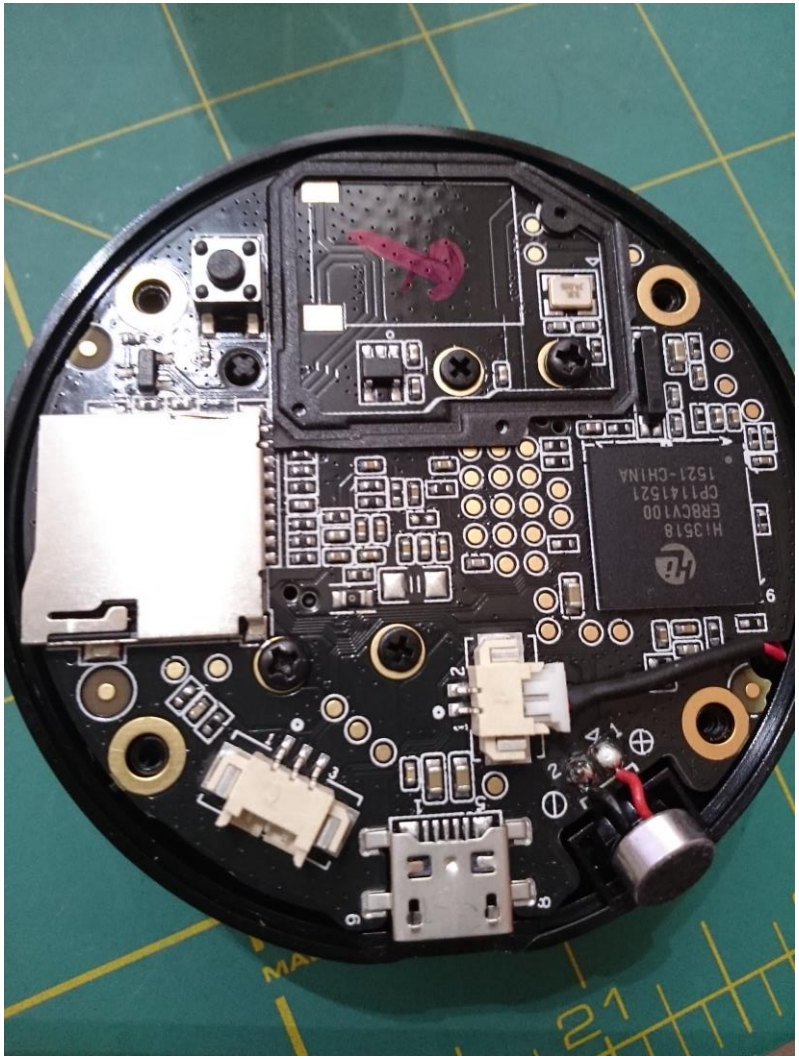
- 17CN 1.8.6.1R_201611191201
- 这个版本是不能被降级的
- 感觉上是设计上的错误造成了降级不成

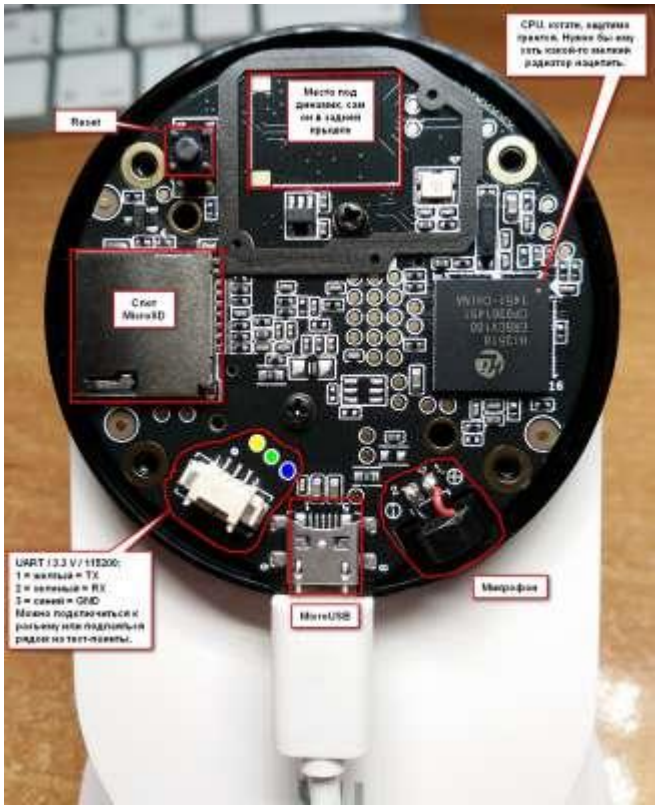
思路



- 电线
- USB TTL
- 网络上流传不同版本的设计
- 亲测 不行

解开谜团





- 寻找GND
- 猜测RX TX
- Multileter 检查

目标

Network settings

> /etc/init.d # cat /home/conf/wpa_supplicant.conf

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
network={
ssid="MY_WIFI_L4H"
scan_ssid=1
proto=WPA RSN
key_mgmt=WPA-PSK
pairwise=CCMP TKIP
group=CCMP TKIP
psk="my_PASSWORD_l4h"
}
```

- 不要APP控制
- 自动启动wifi
- 自动启动telnet
- 自动启动ftp
- 自动启动rtsp

手工配置

Bring up some services

> /etc/init.d # cat S88telnet

```
#!/bin/sh
/home/app/telnetd &
(sleep 10; /home/base/tools/wpa_supplicant -iwlan0 -c/home/conf/wpa_supplicant.conf) &
(sleep 20; /sbin/ifconfig wlan0 192.168.0.100 netmask 255.255.255.0) &
```

> /etc/init.d # cat S89ftp

```
#!/bin/sh
/home/app/tcpsvd -vE 0.0.0.0 21 ftpd -w / &
```

RTSP returns segmentation fault

Fire up IDA pro and look at the RTSP Binary, we found few files requred before it can run, so this is how we fix it.

```
ln -s /tmp/hd1 /home/hd1
ln -s /tmp/hd2 /home/hd2
ln -s /tmp /home/mmap_tmpfs
mkdir /home/jrview
ln -s /home/app/busybox /bin/renice
ln -s /home/lib/libcrypt-0.9.32.1.so libcrypt.so.0
ln -s /home/lib/libstdc\+\+.so.6.0.12 libstdc++.so.6
```

- 运行了telnet ftp 和 RTSP

一个天大的错

```
i2c /dev entries driver
hisi_i2c hisi_i2c.0: Hisilicon [i2c-0] probed!
hisi_i2c hisi_i2c.1: Hisilicon [i2c-1] probed!
hisi_i2c hisi_i2c.2: Hisilicon [i2c-2] probed!
TCP: cubic registered
Initializing XFRM netlink socket
NET: Registered protocol family 17
NET: Registered protocol family 15
lib80211: common routines for IEEE802.11 drivers
Registering the dns_resolver key type
VFS: Mounted root (jffs2 filesystem) on device 31:4.
Freeing init memory: 112K
Kernel panic - not syncing: No init found.  Try passing init= option to kernel. See Linux Documentation/init.txt for
```

- 误杀了内核

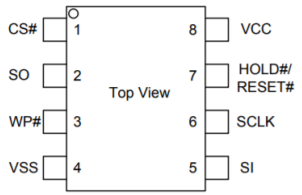
Data Sheet

GD25Q128CxIGx 3.3V Uniform Sector Dual and Quad Serial Flash

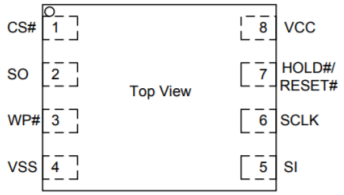
. GENERAL DESCRIPTION

The GD25Q128C(128M-bit) Serial flash supports the standard Serial Peripheral Interface (SPI), and supports ıe Dual/Quad SPI: Serial Clock, Chip Select, Serial Data I/O0 (SI), I/O1 (SO), I/O2 (WP#) and I/O3 (HOLD#/ :ESET#). The Dual I/O data is transferred with speed of 208Mbits/s and the Quad I/O & Quad Output data is ransferred with speed of 320Mbits/s.

Connection Diagram



8-LEAD SOP

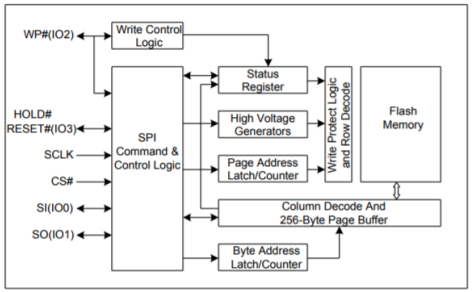


8-LEAD WSON

in Description

Pin Name	I / O	Description
CS#	I	Chip Select Input
SO (IO1)	I/O	Data Output (Data Input Output 1)
WP# (IO2)	I/O	Write Protect Input (Data Input Output 2)
VSS		Ground
SI (IO0)	I/O	Data Input (Data Input Output 0)
SCLK	I	Serial Clock Input
HOLD#/RESET (IO3)	I/O	Hold or Reset Input (Data Input Output 3)
VCC		Power Supply

lock Diagram



- Sdcard是不被读取的

固件分析

XiaoYI Ants unofficial info page

HOME INSTRUCTIONS FIRMWARES BUY A YI

Firmwares

Hardware version v2.1 needs a firmware version 1.8.5.1K or higher!
You can find the how to on the firmware flash instruction page.
Note: flash firmware is at your own risk!

Original for CN hardware

- 1.8.5.1B_201513211614
- 1.8.5.1H_201505211709
- 1.8.5.1J_201507201424
- 1.8.5.1K_201508311131
- 1.8.5.1L_201506291725
- 1.8.5.1M_201512011815
- 1.8.5.1N_201512212009
- 1.8.6.1A_201602241619
- 1.8.6.1B_201603181307

Original for international hardware

- 1.8.5.1N_201601071352

Modified for CN hardware

Additional features are added to this firmwares (RTSP, FTP, telnet, timezone, ...)
How to use the different additional features is described on the instruction page.

- 1.8.5.1B_rtsp
- 1.8.5.1J_easy_boot
- 1.8.5.1K_rtspfix-v3
- 1.8.5.1L_rtspfix-v3
- 1.8.5.1M_rtspfix-v4
- 1.8.6.1B_rtspfix

Branch: master ▾ yi-hack-v3 / src / Create new file

shadow-1 Fixed errors in startup scripts.

..

busybox	Added ability to randomly select the number of proxy servers to downl...
home/yi-hack-v3	Fixed errors in startup scripts.
libwebsockets-plugins	Firmware no longer affected by Xiaomi updates.
libwebsockets	Firmware no longer affected by Xiaomi updates.
proxychains-ng	Firmware no longer affected by Xiaomi updates.
rootfs/etc	Fixed errors in startup scripts.
uClibc	Initial tested version of the firmware for Yi 1080p Dome camera.

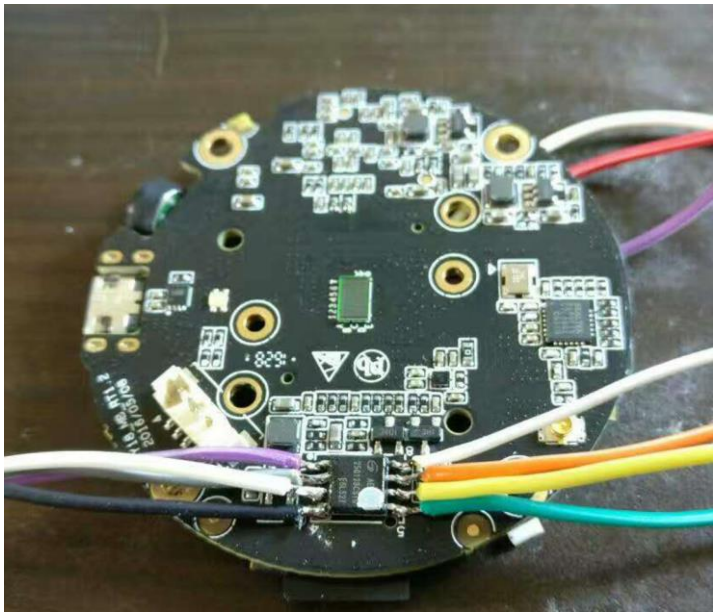
- 缺乏资料的固件

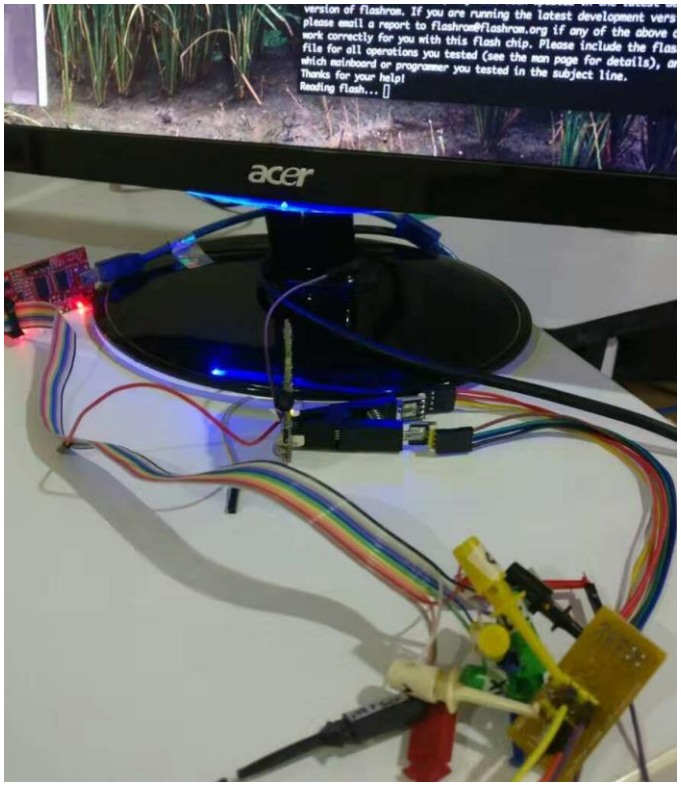
解读dmesg

```
brd: module loaded
Check Flash Memory Controller v100 ...  Found.
SPI Nor(cs 0) ID: 0xc8 0x40 0x18
Block:64KB Chip:16MB Name:"GD25Q128"
SPI Nor total size: 16MB
8 cmdlinepart partitions found on MTD device hi_sfc
8 cmdlinepart partitions found on MTD device hi_sfc
Creating 8 MTD partitions on "hi_sfc":
0x000000000000-0x000000040000 : "boot"
0x000000040000-0x000000050000 : "env"
0x000000050000-0x000000060000 : "conf"
0x000000060000-0x0000001f0000 : "os"
0x0000001f0000-0x000000330000 : "rootfs"
0x000000330000-0x000000fe0000 : "home"
0x000000fe0000-0x000000ff0000 : "vd1"
0x000000ff0000-0x000001000000 : "ver"
ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI) Driver
hiusb-ehci hiusb-ehci.0: HIUSB EHCI
hiusb-ehci hiusb-ehci.0: new USB bus registered, assigned bus number 1
hiusb-ehci hiusb-ehci.0: irq 15, io mem 0x100b0000
hiusb-ehci hiusb-ehci.0: USB 0.0 started, EHCI 1.00
hub 1-0:1.0: USB hub found
hub 1-0:1.0: 1 port detected
i2c /dev entries driver
hisi_i2c hisi_i2c.0: Hisilicon [i2c-0] probed!
hisi_i2c hisi_i2c.1: Hisilicon [i2c-1] probed!
hisi_i2c hisi_i2c.2: Hisilicon [i2c-2] probed!
```

- 缺乏资料的固件

导出固件





- 缺乏资料的固件

拆解固件

Taking Partition Notes

Partition by size, take from the boot log

```
0x000000000000-0x000000040000 : "boot"
0x000000040000-0x000000050000 : "env"
0x000000050000-0x000000060000 : "conf"
0x000000060000-0x0000001f0000 : "os"
0x0000001f0000-0x000000330000 : "rootfs"
0x000000330000-0x000000fe0000 : "home"
0x000000fe0000-0x000000ff0000 : "vd1"
0x000000ff0000-0x000001000000 : "ver"
```

Dump using bus pirate

```
flashrom -p buspirate_spi:dev=/dev/ttyUSB0 -c GD25Q128C -r yicam_night_GD25Q128C.bin -V -f
```

Spliting the image

This is how you split the file according to partition size

```
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_bootloader.bin bs=1 count=$((0x040000))
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_env.bin bs=1 count=$((0x050000-0x040000)) skip=$((
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_conf.bin bs=1 count=$((0x060000-0x050000)) skip=$(
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_os.bin bs=1 count=$((0x1f0000-0x060000)) skip=$((0
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_rootfs.bin bs=1 count=$((0x330000-0x1f0000)) skip=
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_home.bin bs=1 count=$((0xfe0000-0x330000)) skip=$(
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_vd1.bin bs=1 count=$((0xff0000-0xfe0000)) skip=$((
dd if=yicam_night_test_GD25Q128C.bin of=yicam_night_test_GD25Q128C_ver.bin bs=1 count=$((0x1000000-0xff0000)) skip=$(
```

- 缺乏资料的固件

从新封印固件

Mount, Edit and Pad

Look for JFFS mounting tutorial, make all the changes you need Just In case you need padding before mergeing the ROM

```
ruby -e 'print "\xFF" * 393216' >> rootfs_e.jjfs
```

Merging the ROM

```
(dd if=yicam_night_test_GD25Q128C_bootloader.bin ) > yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_env.bin ) >> yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_conf.bin ) >> yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_os.bin ) >> yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_rootfs_e.bin ) >> yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_home.bin ) >> yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_vd1.bin ) >> yicam_full_e.bin
(dd if=yicam_night_test_GD25Q128C_ver.bin ) >> yicam_full_e.bin
```


测试

拆解jffs2

TL;DR

Here's a quick overview of the entire mounting process:

1. Extract the JFFS2 file system image from the U-Boot image:
   `uImage.py -x home`
2. Pad the JFFS2 image to make it work with `block2mtd`:
   `./jffs2.py --pad=0 7518-hi3518-home`
3. Load the kernel modules:
   `modprobe block2mtd mtdblock`
4. Setup the loopback device:
   `losetup /dev/loop0 7518-hi3518-home`
5. Associate loopback device with MTD device
6. Mount the MTD device (finally)

If all this seems tedious, I wrote a `mount-jffs2` shell script that performs steps 3 to 6. You just need to specify the (padded) image file, mount point and block size:

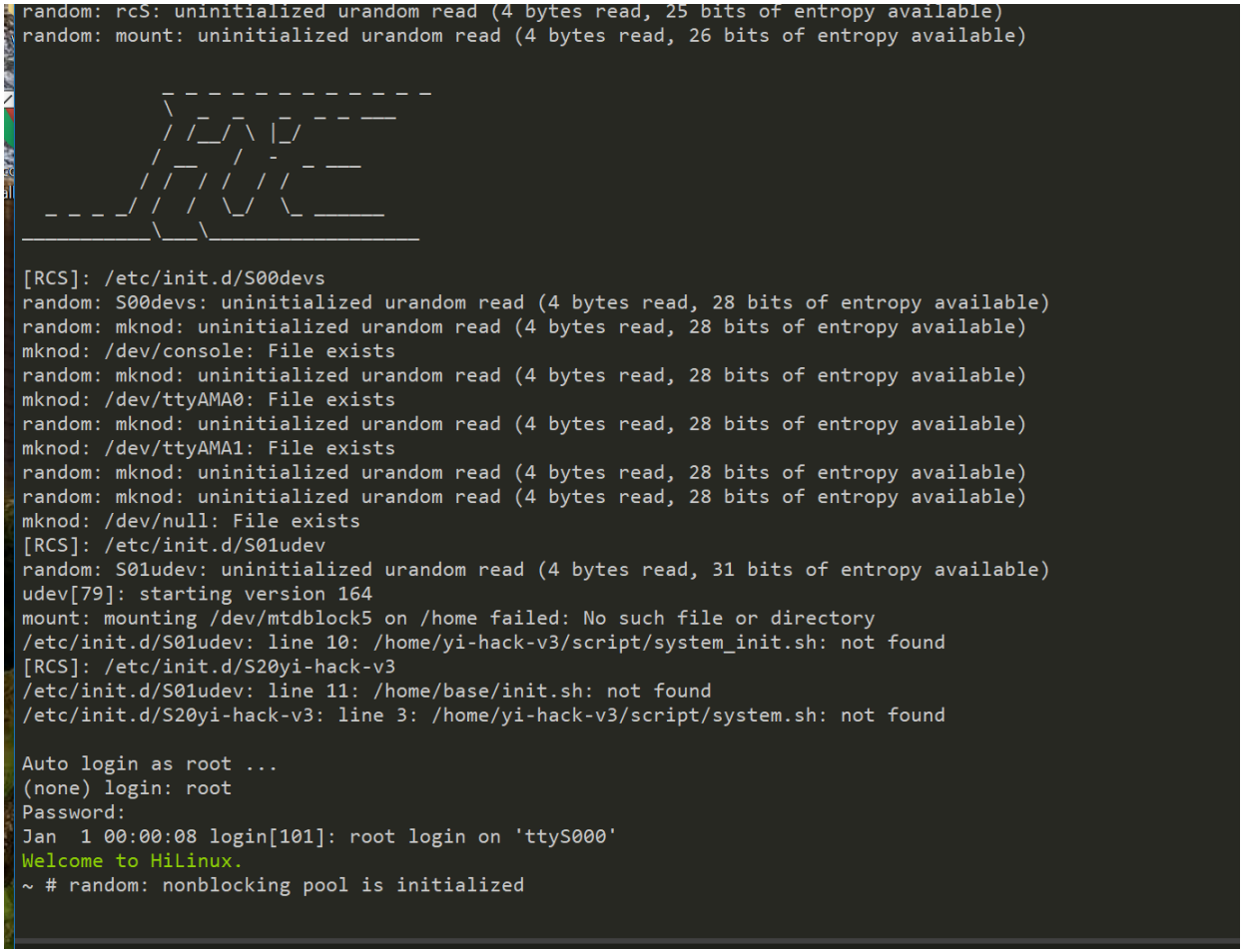
```
./mount-jffs2 7518-hi3518-home /mnt/image 64KiB
```

制造img

```
(23:52:06):xwings@kali32:<~/yicam_home_720p/yi-hack-v3/rootfs_mount>
(117)$ ls -alF
total 60
drwxr-xr-x 15 root   root   4096 Jan  1  1970 ./
drwxr-xr-x  5 xwings xwings 4096 Aug 15 23:11 ../
drwxr-xr-x  2 root   root   4096 Jul  2 22:34 bin/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 dev/
drwxr-xr-x  4 root   root   4096 Jul  2 22:24 etc/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 home/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 lib/
lrwxrwxrwx  1 root   root     11 Jul  2 22:34 linuxrc -> bin/busybox*
drwxr-xr-x  3 root   root   4096 Jul  2 22:24 mnt/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 proc/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 root/
drwxr-xr-x  2 root   root   4096 Jul  2 22:34 sbin/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 sys/
drwxr-xr-x  2 root   root   4096 Jul  2 22:24 tmp/
drwxr-xr-x  4 root   root   4096 Jul  2 22:34 usr/
drwxr-xr-x  3 root   root   4096 Jul  2 22:24 var/
(23:52:08):xwings@kali32:<~/yicam_home_720p/yi-hack-v3/rootfs_mount>
```

- # qemu-img create test.img 1024M
- # mkfs.ext2 –F test.img
- # mount –t ext2 –o loop,rw test.img /mnt/test
- Copy all files
- umount

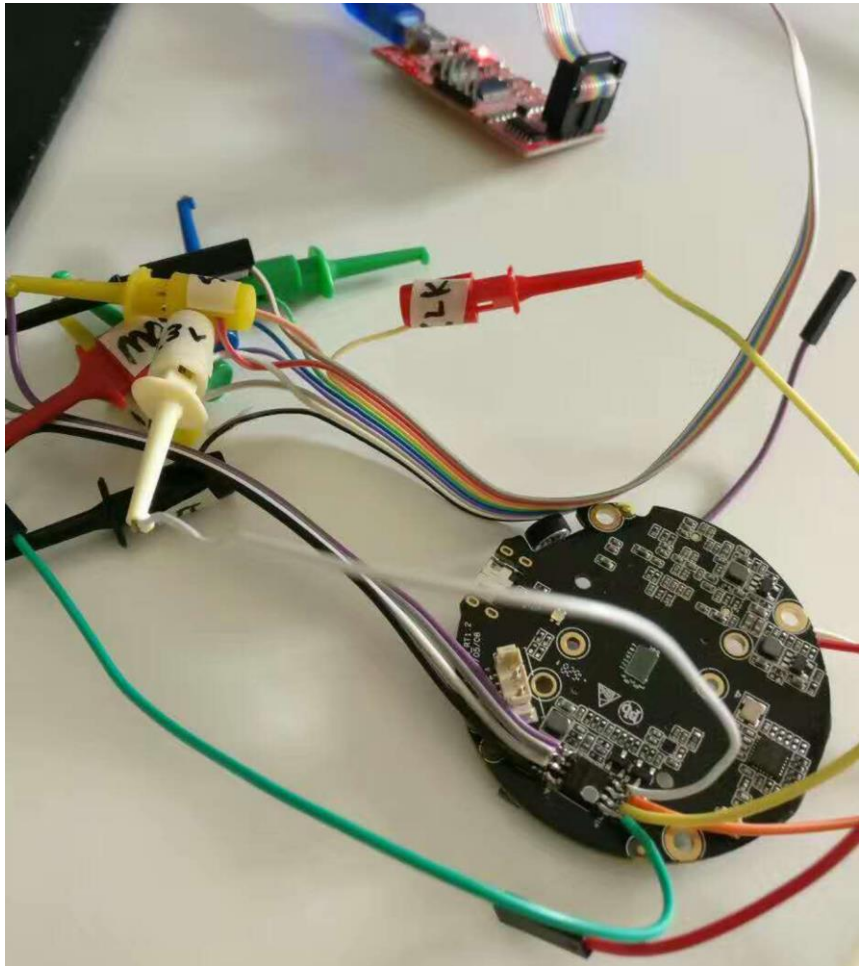
拆解jffs2



- /home/xwings/qemu-2.9.0/arm-softmmu/qemu-system-arm -cpu arm1176 -M versatilepb -kernel /home/xwings/yicam_home_720p/testrun/kernel-qemu-4.4.34-jessie -append "console=ttyAMA0 root=/dev/sda rootfstype=ext2 rw" -hda /home/xwings/yicam_home_720p/yi-hack-v3/rootrootfs.img -nographic


封印

还原



- 缺乏资料的固件

复习 – 路由器 （为何）

TP-Link WDR6500 V6

快速搜索

搜索“TL-WDR6500”得到的结果：

文档中心

- [云路由器] 手机APP设置路由器上网
- [TL-WDR6500千兆版] 介绍、设置、使用、问题解答综合指南
- [TL-WDR6500千兆版] 如何设置路由器上网？
- [TL-WDR6500千兆版] 如何当作无线交换机使用？

下载中心

- TL-WDR6500千兆版 V1.0_170725标准版
- TL-WDR6500 V6.0_1.0.0详细配置指南
- TL-WDR6500千兆版 V1.0_1.0.0详细配置指南
- TL-WDR6500 V6.0_170503标准版

- 刷机其实很简单 ？
- OpenWRT/LEDE 只有V2
- 我们可以百度一下

不可能的任务

TP-LINK TL-WDR3228 v2	QCA9558+AR8035	6401K
TP-Link TL-WDR3320 v2	AR9344+AR9287	6401K
TP-LINK TL-WDR3500 v1	AR9344+AR9582	6401K
TP-LINK TL-WDR3600 v1	AR9344+AR8327N+AR9582	6401K
TP-LINK TL-WDR4300 v1	AR9344+AR8327N+AR9580	6401K
TP-Link TL-WDR4310 v1	AR9344+AR8327N+AR9580	6401K
TP-LINK TL-WDR4900 v2	QCA9558+AR8327N+AR9580	6721K
TP-LINK TL-WDR6500 v2	QCA9561+QCA9882	6721K
TP-LINK TL-WDR6500 v6	QCA9561+QCA9886	6913K
TP-LINK TL-WR1041N v2	AR9342+AR8327N	6401K
TP-LINK TL-WR1041N v2	QCA9558+AR8327N	6401K
TP-LINK TL-WR2543N/ND	AR7242+RTL8367R	6401K
TP-Link TL-WR741N/ND v4	AR9331	6401K
TP-Link TL-WR841N/ND v5	AR7240	6401K
TP-LINK TL-WR841N/ND v7	AR7241	6401K
TP-LINK TL-WR841N/ND v8	AR9341	6401K
TP-LINK TL-WR841N/ND v9	QCA953X	6401K

- RSA签名固件检测干掉了
- V6 干掉了TX？
- 这里的V6是骗人的 ？

V6 的不死breed

Breed Web 恢复控制台

- 系统信息
- 固件更新
- 固件备份
- 频率设置
- 恢复出厂设置
- TP-LINK 设置
- 重启
- 关于

CPU	Qualcomm Atheros QCA956X rev 0
内存	64MB DDR2
Flash	GigaDevice GD25Q64 @ 17MHz (8MB)
以太网	Atheros AR8228/AR8229 rev 1
时钟频率	CPU: 750MHz, DDR: 650MHz, AHB: 250MHz, Ref: 25MHz
编译日期	2017-08-13 [git-f17d10a]
版本	1.1 (r1041)

```
(22:34:27):root@kali32:<~>
(3)# cu -l /dev/ttyUSB0 -s 115200
Connected.

Boot and Recovery Environment for Embedded Devices
Copyright (C) 2017 HackPascal <hackpascal@gmail.com>
Build date 2017-08-13 [git-f17d10a]
Version 1.1 (r1041)

DRAM: 64MB
Platform: Qualcomm Atheros QCA956X rev 0
Board: TP-LINK TL-WDR6500 v2
Clocks: CPU: 750MHz, DDR: 650MHz, AHB: 250MHz, Ref: 25MHz
Flash: GigaDevice GD25Q64 (8MB) on ath79-spi
ag71xx-eth: MAC address is invalid, using default settings.
ag71xx-eth: Using MAC address 00:13:74:00:00:01
Waiting for auto-negotiation complete ... Timed out
eth0: Atheros AR8228/AR8229 rev 1

Network started on eth0, inet addr 192.168.1.1, netmask 255.255.255.0

Press any key to interrupt autoboot ... 0

Unable to locate firmware.

Starting breed built-in shell
```

- 讨论一下到底如何能刷到这里
- Tips: https://breed.hackpascal.net/

Bootable to OpenWRT

No password set!

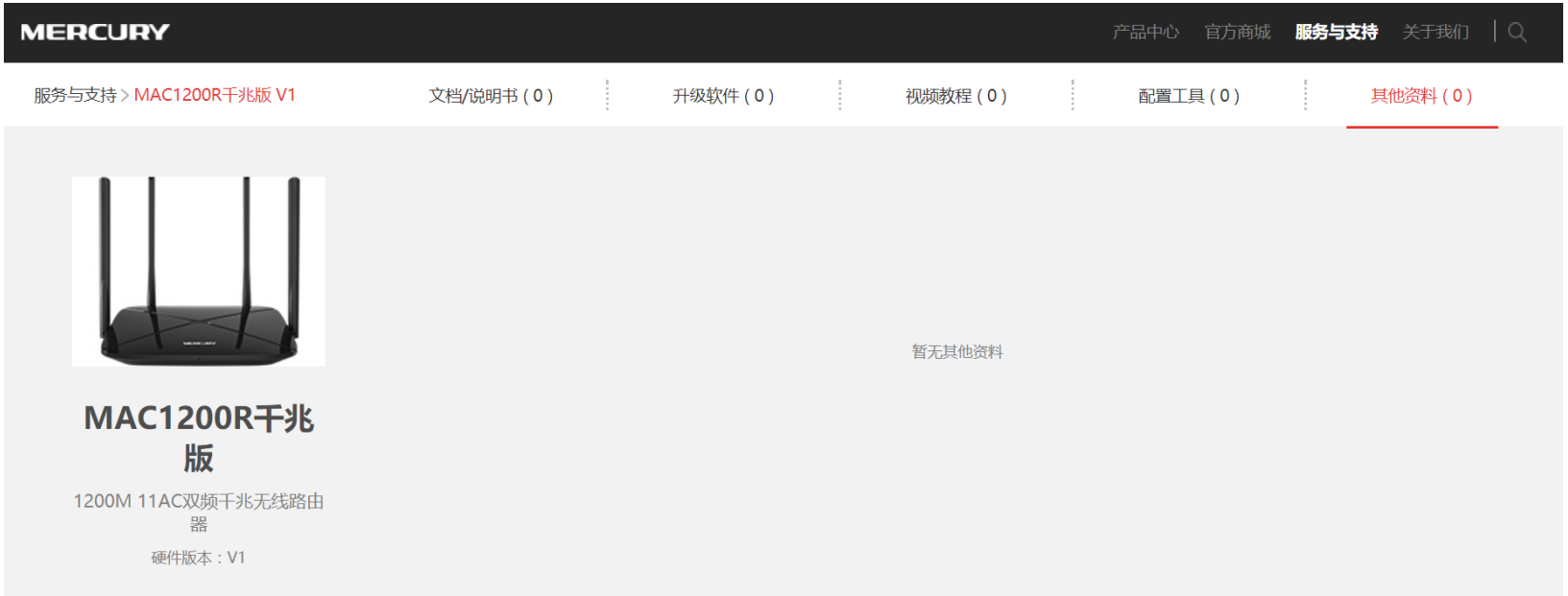
There is no password set on this router. Please configure a root password to protect the web interface and enable SSH.
Go to password configuration...

System Log

```
Wed Jan 18 01:40:10 2017 kern.notice kernel: [    0.000000] Linux version 4.4.14 (qiao@qiao.guoxiaoqiao.cn) (gcc version 5.3.0 (OpenWrt GCC 5.3.0 50082) ) #9
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000] MyLoader: sysp=ed9573bc, boardp=bb069a8a, parts=35f6dce5
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] bootconsole [early0] enabled
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] CPU0 revision is: 00019750 (MIPS 74Kc)
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] SoC: Qualcomm Atheros QCA956X ver 1 rev 0
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Determined physical RAM map:
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000]  memory: 04000000 @ 00000000 (usable)
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Initrd not found or empty - disabling initrd
Wed Jan 18 01:40:10 2017 kern.warn kernel: [    0.000000] No valid device tree found, continuing without
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Zone ranges:
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000]   Normal   [mem 0x0000000000000000-0x0000000003ffffff]
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Movable zone start for each node
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Early memory node ranges
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000]   node   0: [mem 0x0000000000000000-0x0000000003ffffff]
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Initmem setup node 0 [mem 0x0000000000000000-0x0000000003ffffff]
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000] On node 0 totalpages: 16384
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000] free_area_init_node: node 0, pgdat 8040e3d0, node_mem_map 81000000
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000]   Normal zone: 128 pages used for memmap
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000]   Normal zone: 0 pages reserved
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000]   Normal zone: 16384 pages, LIFO batch:3
Wed Jan 18 01:40:10 2017 kern.warn kernel: [    0.000000] Primary instruction cache 64kB, VIPT, 4-way, linesize 32 bytes.
Wed Jan 18 01:40:10 2017 kern.warn kernel: [    0.000000] Primary data cache 32kB, 4-way, VIPT, cache aliases, linesize 32 bytes
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000] pcpu-alloc: s0 r0 d32768 u32768 alloc=1*32768
Wed Jan 18 01:40:10 2017 kern.debug kernel: [    0.000000] pcpu-alloc: [0] 0
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Built 1 zonelists in Zone order, mobility grouping on.  Total pages: 16256
Wed Jan 18 01:40:10 2017 kern.notice kernel: [    0.000000] Kernel command line:  board=TL-WDR6500-v2  console=ttyS0,115200 rootfstype=squashfs,jffs2 noini
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] PID hash table entries: 256 (order: -2, 1024 bytes)
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Dentry cache hash table entries: 8192 (order: 3, 32768 bytes)
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Inode-cache hash table entries: 4096 (order: 2, 16384 bytes)
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Writing ErrCtl register=00000000
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Readback ErrCtl register=00000000
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Memory: 60100K/65536K available (2974K kernel code, 174K rwdata, 756K rodata, 312K init, 200K bss,
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] SLUB: HWalign=32, Order=0-3, MinObjects=0, CPUs=1, Nodes=1
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] NR_IRQS:51
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] Clocks: CPU:750.000MHz, DDR:650.000MHz, AHB:250.000MHz, Ref:25.000MHz
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000000] clocksource: MIPS: mask: 0xffffffff max_cycles: 0xffffffff, max_idle_ns: 5096694524 ns
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.000007] sched_clock: 32 bits at 375MHz, resolution 2ns, wraps every 5726622718ns
Wed Jan 18 01:40:10 2017 kern.info kernel: [    0.008830] Calibrating delay loop... 373.55 BogoMIPS (lpj=1867776)
```

- 缺乏WiFi 2.4G
- 缺乏Wifi 5G

MAC1200R Gigabit Edition V1



- 缺乏资料的一整个东西

Breed MT7620A

水星mac1200r千兆版v1.0拆机应该是首拆 [复制链接]

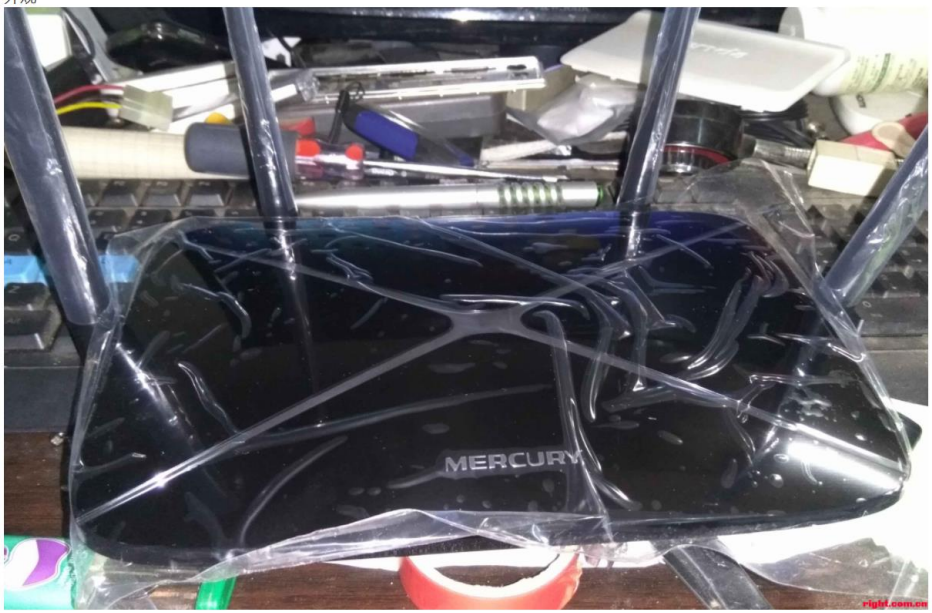
发表于 2017-6-6 00:05 | 只看该作者 | 只看大图

本帖最后由 shwpds 于 2017-6-6 23:10 编辑

水星mac1200r千兆版v1.0拆机应该是首拆

cpu是MT7620a，千兆版有独立SW，胶粘着没看到型号，明天找个薄刀片，切开看看，我会首先发布出来。。。
5g缩水了，功率不高，待机3w，8+64m配置，下面上图，图片以后有TTL启动的数据：

外观



- 讨论一下应该步骤？

GD25Q256SIG

GD25Q64C规格书

蓝枫0107 上传于 2015-02-15 ★5.0分(高于99%的文档) 5770 273 简

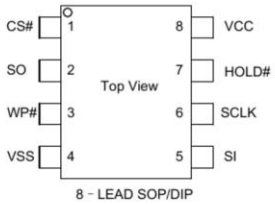
★ 优质文档

3.3V Uniform Sector
Dual and Quad Serial Flash

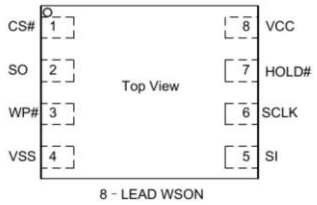
2. GENERAL DESCRIPTION

The GD25Q64C (64M-bit) Serial flash supports the standard Serial Peripheral Interface (SPI), and supports the Dual/Quad SPI: Serial Clock, Chip Select, Serial Data I/O0 (SI), I/O1 (SO), I/O2 (WP#), and I/O3 (HOLD#). The Dual I/O data is transferred with speed of 240Mbits/s and the Quad I/O & Quad output data is transferred with speed of 480Mbits/s.

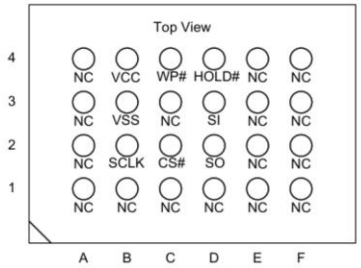
CONNECTION DIAGRAM



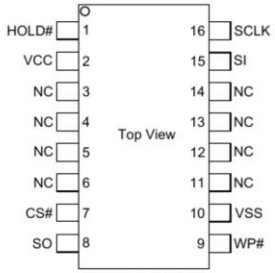
8 - LEAD SOP/DIP



8 - LEAD WSON



24-BALL TFBGA



16-LEAD SOP

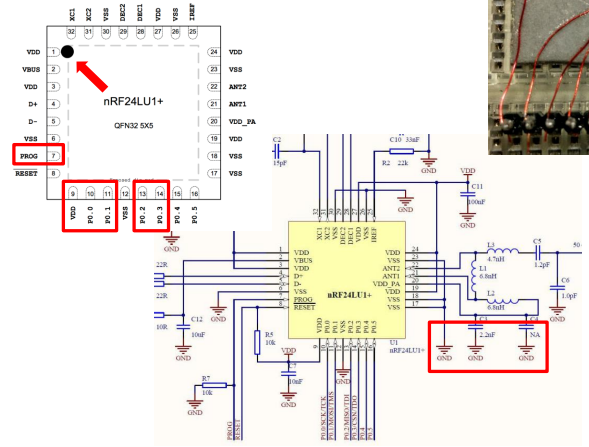
- 直接刷
- 如何链接
- 夹子
- 清空 “0xFF”
- 写入 与 认证
- 如何破解RSA签名的固件

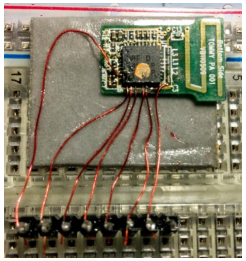

学习到了什么

Data Sheet

Identifying the Pins

```
#!/usr/bin/perl -w

# Simple perl script to drive the Bus Pirate and unbrick your CrazyRadio dongle.
# Adapted (sorta) from the Bus Pirate example script and mbed NRF24LU1+ flasher projects:
#  http://code.google.com/p/the-bus-pirate/source/browse/trunk/scripts/SPIeeprom.pl
#  http://mbed.org/users/mux/code/nrflash
#
# This script uses the aux output on the Bus Pirate as the PROG pin on the CrazyRadio's NRF24LU1+ chip.
#
# Electrical connections are as follows:
#
#  Bus Pirate      CrazyRadio
#  ==========================
#  MOSI ()     ->  MOSI   (6)
#  MISO ()     ->  MISO   (8)
#  SCK  ()     ->  SCK    (4)
#  CS   ()     ->  CS     (10)
#  AUX  ()     ->  PROG   (2)
#  3V3  ()     ->  3V3    (5)
#  GND  ()     ->  GND    (9)

use strict;
use feature 'say';
use Getopt::Long;
use Device::SerialPort;
use Time::HiRes qw/usleep/;

use constant {
    WREN        => "\x06",
    WRDIS       => "\x04",
    RDSR        => "\x05",
    WRSR        => "\x01",
    READ        => "\x03",
    PROGRAM     => "\x02",
    ERASE_PAGE  => "\x52",
    ERASE_ALL   => "\x62",
    RDFPCR      => "\x89",
    RDISMB      => "\x85",
    ENDEBUG     => "\x86",
    RDYN        => "\x10",
    FLASH_LEN   => 32768,

    BP_CS       => "\x01",
    BP_AUX      => "\x02",
    BP_PULLUP   => "\x04",
    BP_POWER    => "\x08",
};

my %opts;
my $port;
my $time = 500;
my $status_byte;
my $return;
```





- MOSI - Pin 11
- MISO - Pin 13
- SCK - Pin 10
- CS - PIN 14
- AUX – Pin 7
- 3V3 - Pin 1
- GND – Any GND

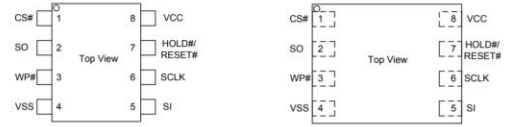
GD25Q128CxIGx 3.3V Uniform Sector Dual and Quad Serial Flash

http://www.elm-tech.com

. GENERAL DESCRIPTION

The GD25Q128C(128M-bit) Serial flash supports the standard Serial Peripheral Interface (SPI), and supports he Dual/Quad SPI: Serial Clock, Chip Select, Serial Data I/O0 (SI), I/O1 (SO), I/O2 (WP#) and I/O3 (HOLD#/ :ESET#). The Dual I/O data is transferred with speed of 208Mbits/s and the Quad I/O & Quad Output data is ansferred with speed of 320Mbits/s.

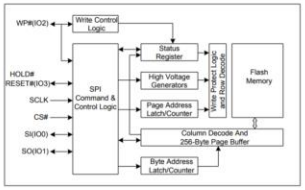
'onnection Diagram



'in Description

Pin Name	I / O	Description
CS#	I	Chip Select Input
SO (IO1)	I/O	Data Output (Data Input Output 1)
WP# (IO2)	I/O	Write Protect Input (Data Input Output 2)
VSS		Ground
SI (IO0)	I/O	Data Input (Data Input Output 0)
SCLK	I	Serial Clock Input
HOLD#/RESET (IO3)	I/O	Hold or Reset Input (Data Input Output 3)
VCC		Power Supply

lock Diagram

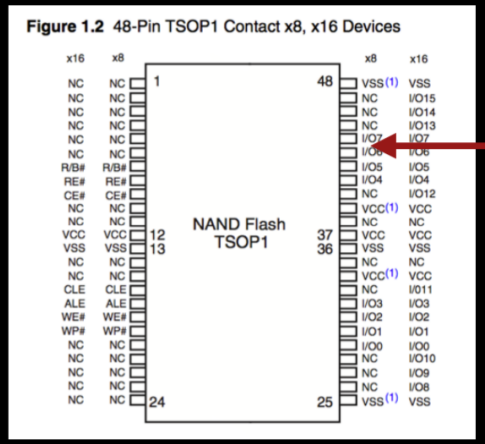


- Data Sheet 永远都是你的答案

小tips 0

Scenario #1: Exploitable U-Boot Configuration

1. No JTAG.
2. Homegrown “secure” boot
3. Try to load and boot kernel #1
4. Try to load and boot kernel #2
5. If that fails then… return to U-Boot prompt!

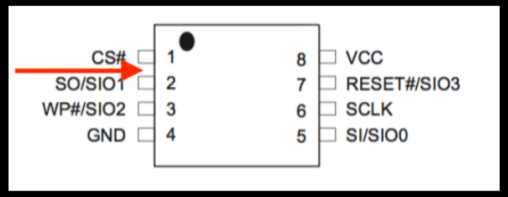


- 进入调试模式
- 找到shell

小tips 1

Scenario #2: Exploitable Init Configuration

- `/bin/init` reads `/etc/inittab`
- `/bin/init` runs `/etc/rc`
- `/etc/rc` starts application in the foreground
- Application grabs console and presents a login prompt with credentials we don't know
- BUT… if the application fails to load then `/bin/init` runs `/bin/sh`



- 避开init
- 找到shell


下课